505-41-30

Interface Control Document Between EOSDIS Core System (ECS) and the Version 0 System for Interoperability

October 1997

Revision B

National Aeronautics and
Space Administration

Goddard Space Flight Center
Greenbelt, Maryland

INTERFACE CONTROL DOCUMENT
between the
EOSDIS Core System (ECS) and the
Version 0 System for Interoperability

Reviewed by:

Greg Hunolt
DAAC System/Science Operations Manager
GSFC - Code 423

Date

Candace Carlisle
Interface Manager
GSFC - Code 505

Date

Dawn Lowe
Science Systems Development Manager
GSFC Code 423

Date

Approved by:

Arthur F. Obenschain
ESDIS Project Manager
GSFC - Code 423

Date

GODDARD SPACE FLIGHT CENTER

GREENBELT, MARYLAND

This page intentionally left blank.

Preface

This revision of the Interface Control Document Between the EOSDIS Core System (ECS) and the Version 0 System brings the document content into alignment with the allocation of Release B functionality between Releases B.0 and B.1. It incorporates corrections but not enhancements to the V0 ODL message protocol forms and keyword definitions, updates information on valids transfer, and adds group and keyword information specifically needed for valids transfer.

This document is a formal contract deliverable with an approval code 1. It requires Government review and approval prior to acceptance and use and changes also require Government approval prior to acceptance and use. Changes to this document shall be made by document change notice (DCN) or by complete revision.

This document is under ESDIS configuration control. Any questions should be addressed to:
ESDIS Project Configuration Management Office
Code 423
Goddard Space Flight Center NASA
Greenbelt, MD 20771

This page intentionally left blank.

Abstract

The Distributed Active Archive Centers (DAACs) support the science community by supplying, via ECS and the EOSDIS Version 0 (V0) Systems, data archive, distribution, information management and product generation services for a wide range of data sets related to global change research. At ECS Release B, Level 3 two-way catalog interoperability between ECS and the EOSDIS V0 Systems will enable users of either system to search, browse, and order data products made available by the other system. This Interface Control Document between the EOSDIS Core System (ECS) and the Version 0 System defines the design of each catalog interoperability interface between ECS and the EOSDIS V0 Information Management System (IMS).

Specifically, this ICD defines the data flows that exist between ECS and the EOSDIS V0 IMS for two-way catalog interoperability, including the following: directory search request/results, guide search request/results, inventory search request/results, browse request/results, product request/result, acknowledge and quit. Formats of Valids Files that must be transferred to support interoperability are also included.

This revision of the ICD brings the document content into alignment with the allocation of Release B functionality between Releases B.0 and B.1. It incorporates corrections but not enhancements to the V0 ODL message protocol forms and keyword definitions, updates information on valids transfer, and adds group and keyword information specifically needed for valids transfer.

This ICD is consistent with the ECS-to-V0 System interface requirements, as described in the Earth Science Data and Information System (ESDIS) Project -- Level 2 Requirements, the Functional and Performance Requirements Specification for the Earth Observing System Data and Information System (EOSDIS) Core System (ECS Level 3 requirements), and the Interface Requirements Document (IRD) Between EOSDIS Core System (ECS) and the V0 System.

Keywords: B.0, B.1, B0SOT, browse, catalog, DAAC, dictionary, directory, guide, HTTP, interface, interoperability, inventory, JEST, ODL, object, order, product, release, science, search, system, valids, Version 0, V0, WAIS

This page intentionally left blank.

Change Information Page

ISSUE	DATE	PAGES AFFECTED	DESCRIPTION
Baseline	02/20/96	All	CCR 505-41-37-002
Revision A	06/24/96	All	CCR-505-41-30-002-B
Revision B	10/28/97	All	CCR 505-41-30-003-A

This page intentionally left blank.

List of Affected Pages

Page No.	Revision	Page No.	Revision	Page No.	Revision	Page No.	Revision
Title	Revision B	3-4	Revision B	4-24	Revision B	A-10	Revision B
i	Revision B	4-1	Revision B	4-25	Revision B	A-11	Revision B
ii	Revision B	4-2	Revision B	4-26	Revision B	A-12	Revision B
iii	Revision B	4-3	Revision B	5-1	Revision B	A-13	Revision B
iv	Revision B	4-4	Revision B	5-2	Revision B	A-14	Revision B
v	Revision B	4-5	Revision B	5-3	Revision B	A-15	Revision B
vi	Revision B	4-6	Revision B	5-4	Revision B	A-16	Revision B
vii	Revision B	4-7	Revision B	5-5	Revision B	A-17	Revision B
viii	Revision B	4-8	Revision B	5-6	Revision B	A-18	Revision B
ix	Revision B	4-9	Revision B	5-7	Revision B	A-19	Revision B
x	Revision B	4-10	Revision B	5-8	Revision B	A-20	Revision B
xi	Revision B	4-11	Revision B	5-9	Revision B	A-21	Revision B
xii	Revision B	4-12	Revision B	5-10	Revision B	A-22	Revision B
xiii	Revision B	4-13	Revision B	5-11	Revision B	A-23	Revision B
xiv	Revision B	4-14	Revision B	5-12	Revision B	A-24	Revision B
1-1	Revision B	4-15	Revision B	A-1	Revision B	A-25	Revision B
1-2	Revision B	4-16	Revision B	A-2	Revision B	A-26	Revision B
2-1	Revision B	4-17	Revision B	A-3	Revision B	AB-1	Revision B
2-2	Revision B	4-18	Revision B	A-4	Revision B	AB-2	Revision B
2-3	Revision B	4-19	Revision B	A-5	Revision B		
2-4	Revision B	4-20	Revision B	A-6	Revision B		
3-1	Revision B	4-21	Revision B	A-7	Revision B		
3-2	Revision B	4-22	Revision B	A-8	Revision B		
3-3	Revision B	4-23	Revision B	A-9	Revision B		

This page intentionally left blank.

Contents

Preface

Abstract

1. Introduction

2. Related Documents

3. Interface Overview

4. Data Flow Descriptions

5. Dependent Valids

Appendix A. ODL Message Keywords (Objects)

Abbreviations nd Acronyms

Figures

This page intentionally left blank.

1. Introduction

1.1 Identification

This Interface Control Document (ICD), Contract Data Requirements List (CDRL) Item 029 whose requirements are specified in Data Item Description (DID) 209/SE1, is a required deliverable under the Earth Observing System Data and Information System (EOSDIS) Core System (ECS) Contract (NAS5-60000).

1.2 Scope

This Interface Control Document (ICD) defines the system interfaces that exist between ECS and the EOSDIS V0 System Information Management System (IMS) for Level 3 catalog interoperability. ECS Releases are keyed to mission support: Release B.0 provides support for EOS AM-1 and Landsat 7 operations, as well as SAGE III operations. Release B.1 supports ADEOS II operations and DAO science and adds full science data processing capability. At Release B.0, two-way V.0 interoperability is achieved via the ECS's V0 gateway for V0 access to ECS and via a direct client interface to the V0 system for ECS users. At Release B.1, the V0 gateway will provide two-way interoperability. Releases C & D will provide evolutionary enhancements to the ECS services provided in the earlier Releases.

This ICD does not address internetworking for V0-to-ECS catalog interoperability--this topic is addressed in each of the ECS to DAAC ICDs. Furthermore, interfaces for the migration of V0 data sets are not addressed herein.

The Earth Science Data and Information System (ESDIS) Project has responsibility for the development and maintenance of this ICD. Any changes in the interface requirements must be agreed to and assessed at the ESDIS Project Level. This ICD will be approved under the signature of the ESDIS Project Manager.

This document reflects the ECS technical baseline, maintained by the ECS Configuration Control Board in accordance with ECS technical direction (see Section 2.2).

1.3 Purpose and Objectives

This document is written to formalize the interpretation and general understanding of the interfaces between ECS and the V0 System IMS. This document provides a point of mutual control of external interface definitions via the ESDIS Configuration Control Board (CCB).

1.4 Status and Schedule

This is the final ICD for the ECS-V0 System catalog interoperability interfaces which will be implemented in ECS. This ICD has been submitted as an ECS Project CCB approval Code 1 document. It has been designated to be under full Government CCB control. Changes may be submitted for consideration by Contractor and Government CCBs under the normal change process at any time.

1.5 Organization

Section 1 provides information regarding the identification, scope, purpose and objectives, and organization of this document.

Section 2 is a listing of the related documents, which were used as a source of information for this document.

Section 3 is an overview of the two-way catalog interoperability interfaces between the ECS and the Version 0 System. Specifically, this section describes the purpose of the catalog interoperability interfaces and a high level description of the data flows (as an introduction to the detailed information provided in Section 4). This section also includes two context diagrams.

Section 4 includes the following information:

- A definition of the Object Description Language (ODL) Message Normalization Form (i.e., Group Structure) is provided for each applicable message. In addition, a brief discussion of the ODL conventions is provided in the context of an example.
- The detailed definition of each data interface (i.e., message exchanged) between the EOSDIS V0 system and the ECS.

Section 5 explains dependent valids transfer and gives the Valids file formats.

Appendix A provides a detailed definition of each of the ODL keywords used in the ODL Message Normalization Forms identified in Section 4 as well as those required for furnishing dependent valids information as described in Section 5.

Abbreviations and Acronyms contains an acronym list.

2. Related Documents

2.1 Parent Documents

The following are parent documents from which this document's scope and content derive:

193-208-SE1-001	Methodology for Definition of External Interfaces for the ECS Project
301-CD-002-003	System Implementation Plan for the ECS Project
423-10-01-1	Goddard Space Flight Center, Earth Science Data and Information System (ESDIS) Project Level 2 Requirements, EOSDIS Core System (ECS), Volume 1
423-10-01-5	Goddard Space Flight Center, Earth Science Data and Information System (ESDIS) Project Level 2 Requirements, Volume 5: EOSDIS Version 0
423-41-01	Goddard Space Flight Center, EOSDIS Core System (ECS) Statement of Work
423-41-02	Goddard Space Flight Center, Functional and Performance Requirements Specification for the Earth Observing System Data and Information System (EOSDIS) Core System (ECS)
505-10-20	System Interface Control Plan for the Earth Science Data and Information System (ESDIS Project)
505-41-11	Goddard Space Flight Center, Interface Requirements Document Between EOSDIS Core System (ECS) and the Version 0 System

2.2 Applicable Documents

The following documents are referenced herein and are directly applicable to this document. In the event of conflict between any of these documents and this ICD, this document shall take precedence. Please note that Internet links cannot be guaranteed for accuracy or currency.

305-CD-023-002	Release B SDPS Data Management Subsystem Design Specification for the ECS Project
313-CD-003-002	Communications and System Management Segment (CSMS) Internal Interface Control Document for the ECS Project, Preliminary
210-TP-001-006	Technical Baseline for the ECS Project

505-41-40	Interface Control Document (ICD) Between the EOSDIS Core System (ECS) and the Goddard Space Flight Center (GSFC) Distributed Active Archive Center (DAAC) for the ECS Project
505-41-39	Interface Control Document (ICD) Between the EOSDIS Core System (ECS) and the Langley Research Center (LaRC) Distributed Active Archive Center (DAAC) for the ECS Project
505-41-42	Interface Control Document Between EOSDIS Core System (ECS) and the Oak Ridge National Laboratory (ORNL) Distributed Active Archive Center (DAAC) for the ECS Project
IMSV0-PD-SD-002	Hughes STX Corporation, Messages and Development Data Dictionary - V0 and Release A Message Passing Protocol Specification, September 1995
IMSV0-PD-SD-002 v2.1	Hughes STX Corporation, Messages and Development Data Dictionary - V0 and ASTER/ECS Message Passing Protocol Specification, September 1997
IMSV0-SW-DE-003	Hughes STX Corporation, EOSDIS IMS Guide Subsystem Design Document, 10/93
540-032	Goddard Space Flight Center, EBnet - Distributed Active Archive Center (DAAC) Interface Control Document (ICD)
none	Ryan, Patrick M., Hughes STX Corporation, A Format for Valids and Keyword Definitions Using ODL, Revision 1.5, October 12, 1994.
none	Hughes STX Corporation, IMS Server Cookbook: Setting Up An IMS Server, undated (ECS library catalog No. LIBO 1641.J)
none	Hughes, STX Corporation, An Overview of Valids Support File Processing in V0 IMS Version 6 and the Release A Search and Order Tool
none	Goddard Space Flight Center, ECS Technical Direction No. 11, "PDR Technical Baseline," 12/6/94
none	Davis, Randy; University of Colorado Laboratory for Atmospheric and Space Physics: User's Guide for the Object Description Language (ODL) Processing Software Library, Release 2.1 DRAFT, 3/13/91 (ECS library catalog No. LIBO 2158)
none	Planetary Data System Standards Reference, Version 3.1, 8/94 (WWW access: http://stardust.jpl.nasa.gov/stdref/stdref.html)

2.3 Information Documents

The following documents, although not directly applicable, amplify or clarify the information presented in this document, but are not binding.

604-CD-001-004	Operations Concept for the ECS Project: Part 1-- ECS Overview
604-CD-002-003	Operations Concept for the ECS project: Part 2B -- Release B

This page intentionally left blank.

3. Interface Overview

The interface between the ECS and the V0 system supports Level 3 two-way catalog interoperability to provide an exchange of data and information. Specifically, this interface supports the search, location and acquisition of data between ECS and the V0 system, providing ECS and V0 users with ready access to the data and services provided by the other system. Searches on seven valid field types are supported for ECS-V0 interoperability: data center id, dataset id, source, sensor, parameter, campaign and processing level.

Figures 3-1 and 3-2 are high level context diagrams for the catalog interoperability interfaces between ECS and the V0 system. Notice that the figures cite two ECS clients. The Release B.0 client (Figure 3-1) will be B0SOT (B.0 Search and Order Tool). For Release B.1 (Figure 3-2), there will be a Web client called JEST (JAVA Earth Science Tool). JEST may be operational during the Release B.0 timeframe, but without full Release B.1 functionality. B0SOT will continue to be supported into the B.1 timeframe concurrently with JEST. The catalog interoperability data flows supported are categorized as follows:

a. directory search request/results – for finding datasets

b. guide search request/results – for obtaining detailed information about datasets

c. inventory search request/results – for locating specific granules within a dataset

d. acknowledge – to acknowledge reception of inventory search results chunk

e. browse requests/results – for enabling the user to retrieve/view representative images, as well as non-image data. There are two browse modes available:

    (1) FTP browse – in response to a browse request submitted through the client, representative images are automatically staged at a designated FTP site, and a message is returned to the user containing sufficient information to allow the user to retrieve the images via FTP

    (2) integrated browse – in response to a browse request, representative images are returned directly to the user via the client for viewing

f. product requests/results – placement of orders for data sets

g. quit – notification of premature termination of a session due to problems; also used at the normal termination of inventory results exchanges of chunks.

On the V0 side of the interface, some of these messages are implemented using Object Description Language (ODL); while others use HTTP GET commands and WAIS queries. (Specific messages utilizing HTTP, WAIS or ODL are identified and described in Section 4.)

V0 users will use the V0 Client to access ECS services at the DAACs. Beginning with Release B.0, the V0 Client will have ODL interfaces with the ECS V0 Gateway, which will translate between ODL messages and the ECS Science Data Server. At Release B.1, V0 users will begin to use the V0 Client to interface directly with the ECS Document Data Server (Guide) using WAIS formatted queries over HTTP protocol. The ECS Document Data Server will handle translation between the WAIS/HTTP queries and its own processes.

At Release B.0, ECS users accessing V0 system services interface directly through the ECS B.0 Search and Order Tool (B0SOT), a client equivalent to the V0 client. B0SOT uses ODL to talk to the V0 DAAC Data Servers, and HTTP and WAIS to talk to the V0 DAAC Guide Servers.

At Release B.1, ECS users accessing V0 system services will interface to V0 through the ECS JAVA Earth Science Tool (JEST) client. The JEST client will interface to the V0 system via the ECS V0 Gateway. To accommodate the interface to the V0 DAAC Data Servers, the V0 Gateway will translate between the ECS and ODL messages. To accommodate the interface to the V0 DAAC Guide Servers, the V0 Gateway will translate between the ECS and HTTP/WAIS.

This document includes the Valids File format definitions required for providing and regularly updating a consistent set of V0 and ECS Valids to the V0 Clients and the ECS. Consolidated Valids information is provided directly to the V0 Client and to the ECS B0SOT by the V0-ECS Science Team. At Release B.1, each V0 DAAC provides an ODL Valids file to ECS to populate the ECS advertising service and data dictionary, allowing ECS JEST client users to run directory searches on V0 data.

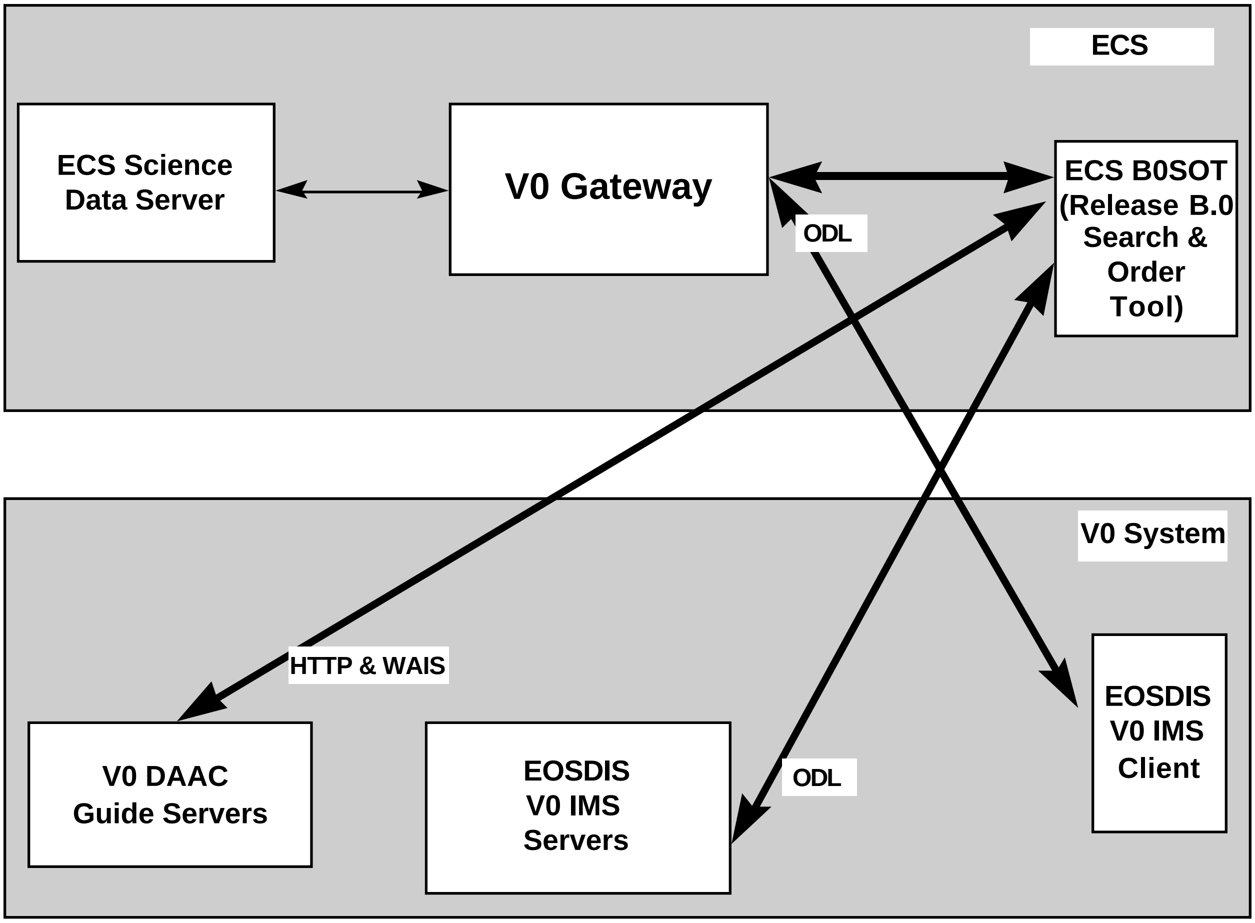


Figure 3-1. ECS/V0 System Interoperability Context Diagram. One-way V0 Gateway interoperability and ECS B0SOT client. ECS Document Data Server not implemented. (Figure 3-2 shows V0 Client access to ECS Document Data Server at ECS Release B.1.)

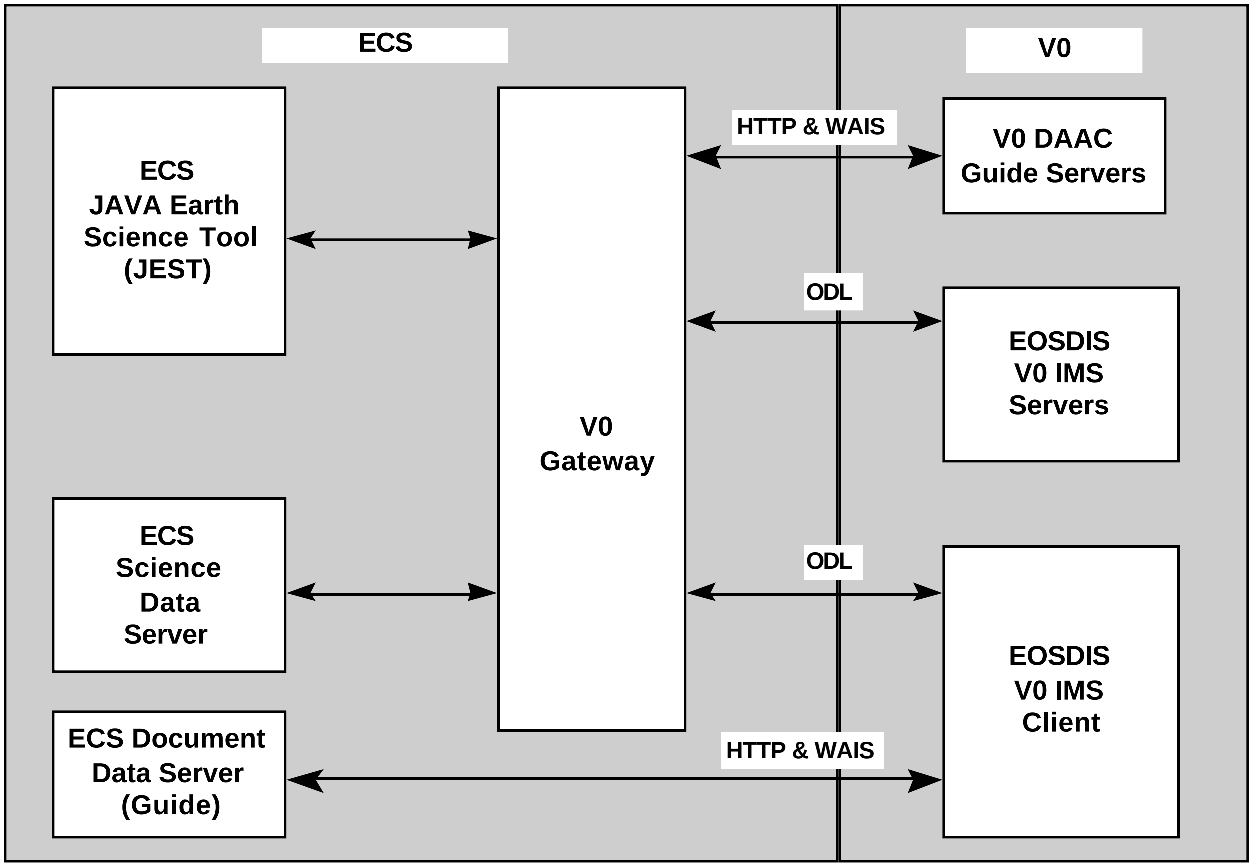


Figure 3-2. ECS/V0 System Interoperability Context Diagram. Two-way V0 Gateway interoperability and ECS JEST client with ECS Document Data Server implemented.

4. Data Flow Descriptions

4.1 General

This section contains the detailed definition of each data interface between ECS and the V0 system that is required to support two-way catalog interoperability. Figures 4-2 through 4-4 diagram the flows for each of these interfaces in the ECS implementation phase contexts explained in Section 3 and presented in Figures 3-1 and 3-2.

With the exception of the guide search request/results messages, all messages use Object Description Language (ODL). (For a description of ODL refer to the User's Guide for the Object Description Language Processing Software Library, Release 2.1 - Draft.) All of these messages are handled by the IMS Kernel (IK) layer [Note: The ECS B0SOT, V0 DAAC Servers and the V0 Gateway contain several software modules, at the communications (lowest) layer, which serve as library routines and are, collectively, referred to as the IK layer].

In the interface descriptions that follow, each interface is treated as a flow between the V0 system and the V0 Gateway, except for the ECS Document Data Server, for which gateway functionality resides in the server. The ODL message is the same for queries in both directions unless otherwise noted.

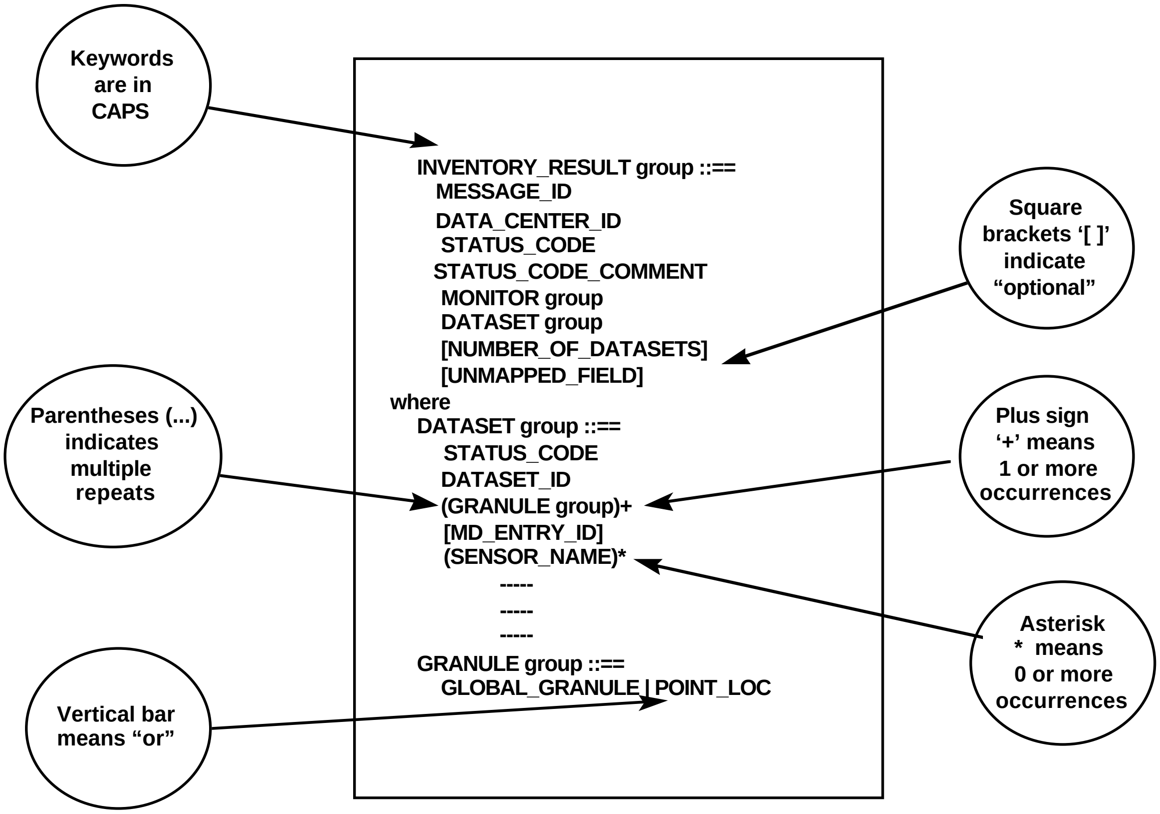
4.2 ODL Conventions

Since many of the above-referenced messages are implemented with ODL, an example of the ODL normalization forms and standardized conventions is provided in Figure 4-1. These standardized conventions, which provide a formal method of describing ODL commands, include the following rules:

- keywords are words that have a special meaning in ODL, itself, and are treated as instructions.
- all keyword are printed in CAPS
- items in square brackets ([ ]) are options.
- items in parentheses (...) indicate that these items may be repeated any number of times
- after the parentheses (...) a single character is given that tells how many occurrences are allowed; i.e.,
    - a '*' means zero or more occurrences
    - a ‘+’ means one or more occurrences
- a vertical bar between items means “or.”
- Each group is further defined down to its keyword components.

In Appendix A, each keyword is defined in terms of the following items of information, as appropriate:

- synopsis (short English-Language description of the keyword),
- parent groups,
- children,
- ODL type; e.g.,
  - integer,
  - real,
  - date,
  - string,
  - aggregate,
  - symbol,
  - sequence string,
  - character string
- maximum (value) length
- possible values.



```
INVENTORY_RESULT group ::==
   MESSAGE_ID
   DATA_CENTER_ID
    STATUS_CODE
   STATUS_CODE_COMMENT
    MONITOR group
    DATASET group
    [NUMBER_OF_DATASETS]
    [UNMAPPED_FIELD]
where
  DATASET group ::==
     STATUS_CODE
    DATASET_ID
    (GRANULE group)+
     [MD_ENTRY_ID]
     (SENSOR_NAME)*
            -----
            -----
            -----
  GRANULE group ::==
    GLOBAL_GRANULE | POINT_LOC
```

Keywords are in CAPS
Square brackets '[ ]' indicate "optional"
Parentheses (...) indicates multiple repeats
Plus sign '+' means 1 or more occurrences
Asterisk * means 0 or more occurrences
Vertical bar means "or"

EXAMPLE Only **EXAMPLE Only**

Figure 4-1. Example of ODL Normalization Form Illustrating Conventions

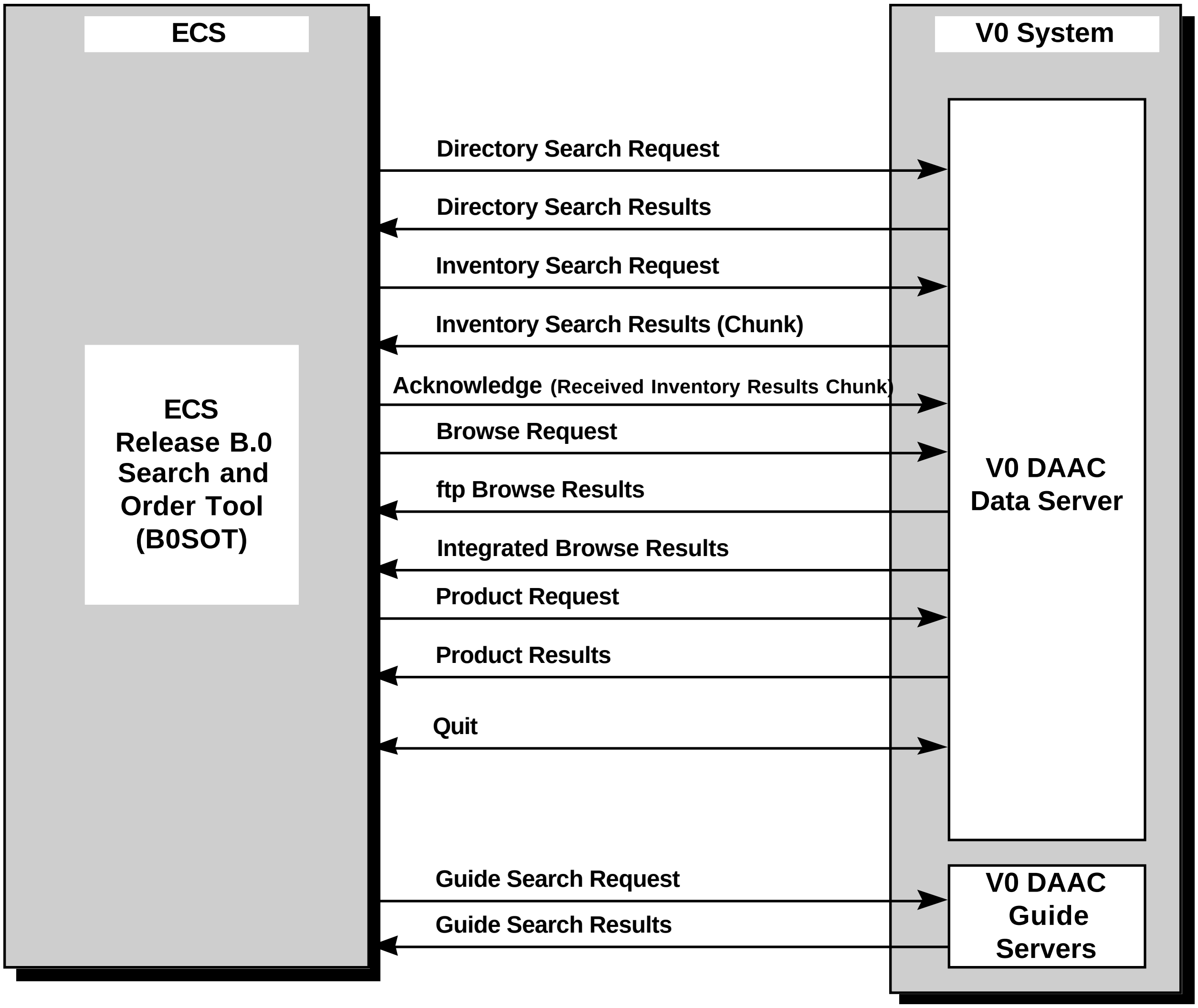


Figure 4-2. Interfaces Between ECS B0SOT Client and V0 IMS Servers

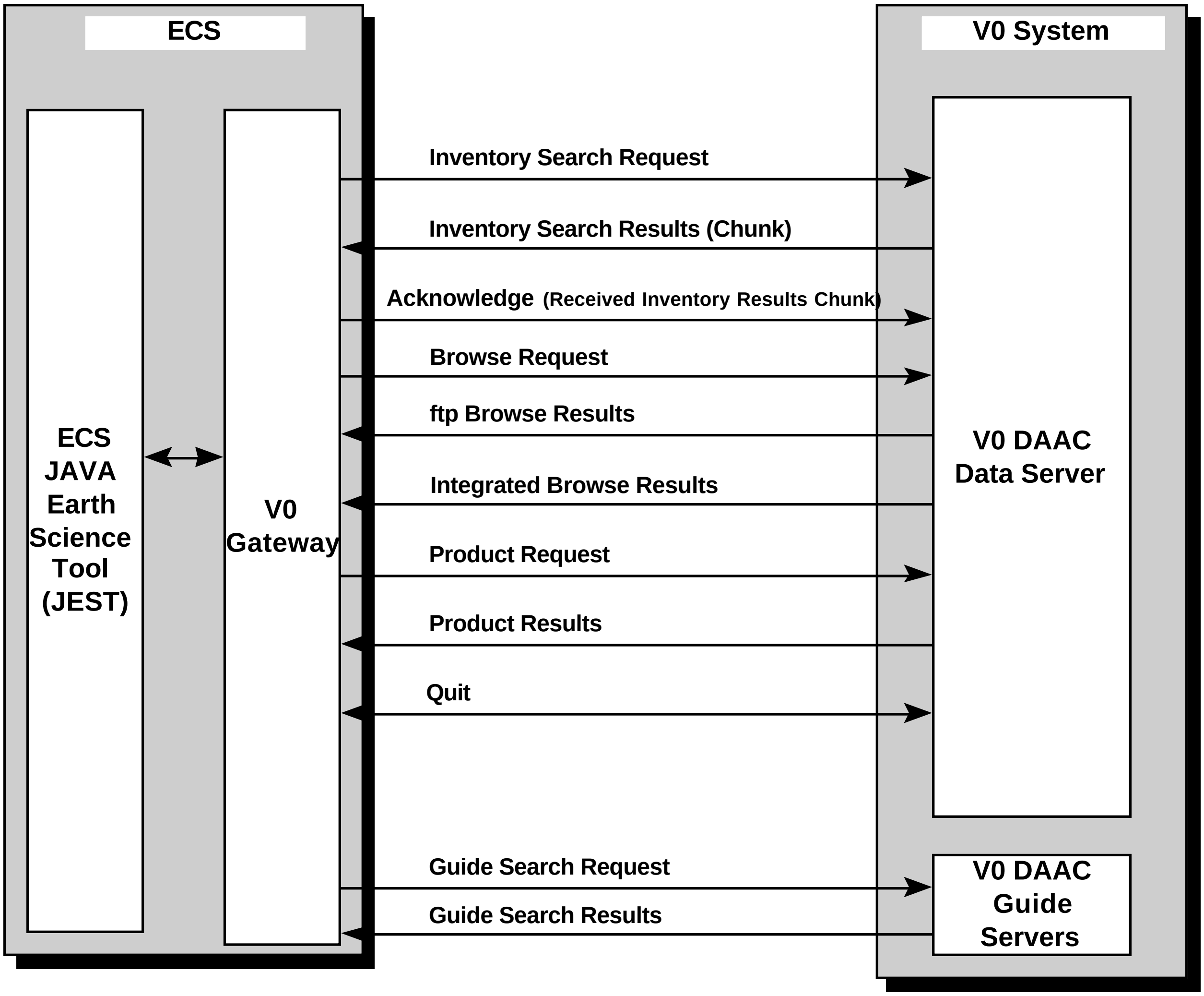


Figure 4-3. Interfaces Between ECS JEST Client (via V0 Gateway) and V0 DAAC Servers

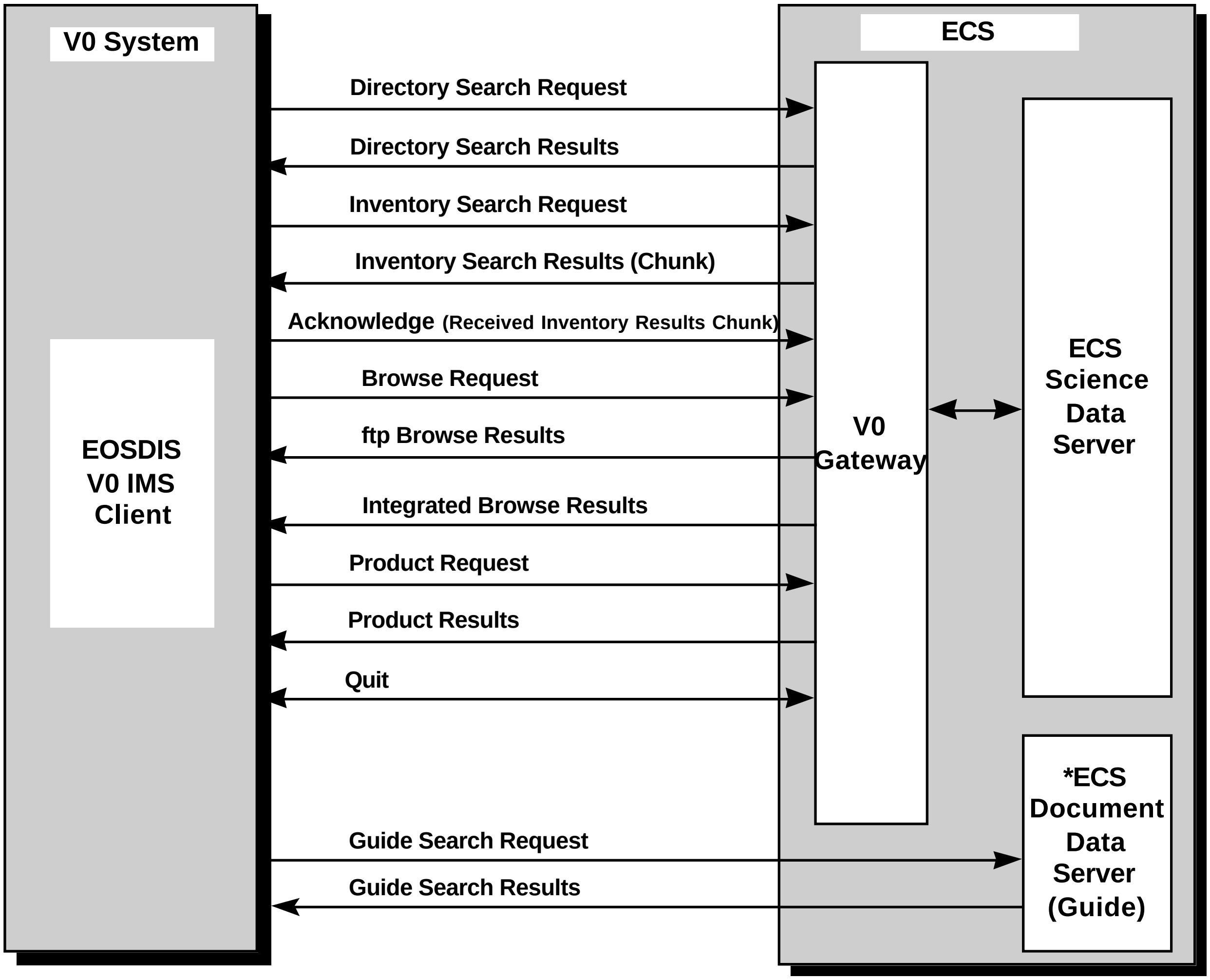


Figure 4-4. Interfaces Between V0 Client (via V0 Gateway) and ECS Servers with ECS Document Data Server Implemented

(*Release B.1)

4.3 Directory Search Request/Results

The purpose of the directory search is to aid the user in making an initial determination of the potential usefulness of various data sets pertinent to some application by searching through descriptions of metadata or data set catalogues which contain high-level information. The directory search provides information on the location of metadata or data set catalogues. The search criteria, specified by the user, include the following searchable attributes: source, sensor, geophysical parameter, dataset name, data center id, campaign, processing level, geographical coordinates (area) and temporal intervals.

A V0 user requesting ECS services submits the directory search request via the V0 Client, which sends the request to the V0 Gateway. The V0 Gateway returns MD_ENTRY_ID keywords to the V0 Client. The V0 Client accesses the Global Change Master Directory (GCMD) using these MD_ENTRY_ID keywords. The GCMD then returns directory information to the V0 Client.

When a V0 directory request is submitted by the ECS B0SOT client, the V0 DAAC Server(s) returns Global Change Master Directory (GCMD) Entry Identifiers (MD_ENTRY_ID). The B0SOT client then uses these keyword identifiers to access the GCMD, which returns directory information.

At Release B.1 an ECS JEST user submits a directory search request for V0 products or services directly to the ECS advertising service via JEST. Thus, for JEST the V0 directory search requirement is implemented without any ODL interface. Section 5 explains how V0 DAACs provide Valids Files to the ECS DAACs in order to support this functionality.

4.3.1 ODL Normalization Form for Directory Search Request

```
DIRECTORY_SEARCH group ::==
    MESSAGE_ID
    [AUTHENTICATOR]
    [ECS_AUTHENTICATOR]
    MONITOR group
    [CAMPAIGN]
    [DATASET_ID]
    [PARAMETER]
    [PROCESSING_LEVEL]
    [SENSOR_NAME]
    [SOURCE_NAME]
    [START_DATE]
    [STOP_DATE]
    [RANGE_LOC group]
    VERSION group

RANGE_LOC group ::==
    NORTH_LATITUDE
    SOUTH_LATITUDE
    EAST_LONGITUDE
    WEST_LONGITUDE

MONITOR group ::==
    TX_CLIENT
    [RX_SERVER]
    [TX_SERVER]
    [RX_CLIENT]
    [SESSION_ID]

VERSION group ::==
    PROTOCOL_VERSION
    SENDER_VERSION
    [IMS_STAFF]
```

4.3.2 ODL Normalization Form for Directory Search Results

```
DIRECTORY_RESULT group ::==
    MESSAGE_ID
    DATA_CENTER_ID
    STATUS_CODE
    [STATUS_CODE_COMMENT]
     (DATASET group)+
    NUMBER_OF_DATASETS
    MONITOR group
    VERSION group

DATASET group ::==
    DATASET_ID
    MD_ENTRY_ID
MONITOR group ::==
    TX_CLIENT
    RX_SERVER
    TX_SERVER
    [RX_CLIENT]
    [SESSION_ID]

VERSION group ::==
    PROTOCOL_VERSION
    SENDER_VERSION
    [IMS_STAFF]
```

4.4 Guide Search Request/Results

The purpose of the guide search is to locate and retrieve guide documents containing detailed information about datasets, based on user-specified keywords or freetext strings; it also allows a user to locate datasets if (s)he decides not to go through a directory or inventory search first. In addition, guide serves to aid a user in understanding the metadata in the search construction process. The search criteria, specified by the user, include the following searchable attributes: source, sensor, geophysical parameter, dataset name, data center id, and campaign. A user can look through the guide documents, follow any hypertext link and display any related documents. The guide documents give detailed information on campaign, source, sensor, data_center, geophysical parameters and dataset.

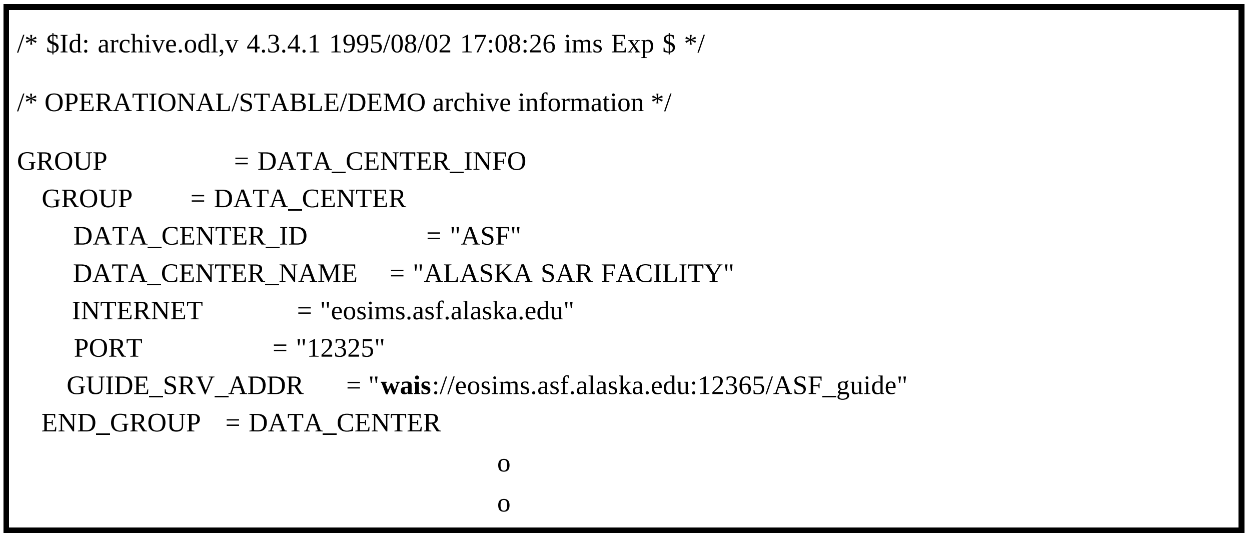
An archive.odl file resides with the V0 Client and the B0SOT client. This file lists the V0 DAAC Guide Servers and (at Release B.1) ECS Document Data Servers, identifying them as WAIS or HTTP servers, respectively. Figure 4-5 depicts an excerpt from the archive.odl file documenting the server address and WAIS protocol for connecting to a V0 DAAC Guide Server. Figure 4-6 depicts an excerpt from the archive.odl file documenting server address and HTTP protocol for connecting to an ECS Document Data Server.

4.4.1 Guide Search Requests from the ECS to V0 Guide Servers

Guide search requests from the ECS are sent as WAIS formatted queries over a modified version of WAIS 0.5 to the V0 DAAC Guide Servers. The ECS B0SOT client has direct WAIS interfaces with the V0 DAAC Guide Servers. Guide searches from the ECS JEST client are converted to the WAIS format by the V0 Gateway.

4.4.1.1 Guide Search Request

The valids information in the ECS B0SOT or, for requests from JEST, in the ECS Data Dictionary, determines which guide servers the guide query is sent to. A separate search message is created for each destination server by B0SOT or, for JEST, by the V0 Gateway on the basis of valids information. The fields and values used in the guide search queries are an encoded version of those used in search and order fields and values used for the Directory and Inventory search functions in the ECS Clients. There is a one-to-one correspondence between these which is automatically handled by the B0SOT or V0 Gateway. The WAIS servers at the V0 DAACs receive the guide queries and perform the search using a WAIS index.



```
/* $Id: archive.odl,v 4.3.4.1 1995/08/02 17:08:26 ims Exp $ */

/* OPERATIONAL/STABLE/DEMO archive information */

GROUP              = DATA_CENTER_INFO
  GROUP       = DATA_CENTER
     DATA_CENTER_ID             = "ASF"
     DATA_CENTER_NAME    = "ALASKA SAR FACILITY"
     INTERNET           = "eosims.asf.alaska.edu"
     PORT                = "12325"
    GUIDE_SRV_ADDR      = "wais://eosims.asf.alaska.edu:12365/ASF_guide"
  END_GROUP   = DATA_CENTER
                                             o
                                             o
```

Figure 4-5. Example Excerpt from archive.odl File Documenting Server Address and WAIS Protocol for Connecting to a V0 DAAC Guide Server (bold emphasis added).

```
/* $Id: archive.odl,v 4.3.4.1 1995/08/02 17:08:26 ims Exp $ */

/* OPERATIONAL/STABLE/DEMO archive information */

GROUP               = DATA_CENTER_INFO

  GROUP      = DATA_CENTER
     DATA_CENTER_ID              = "[GSFC_ECS]"
     DATA_CENTER_NAME    = "GODDARD SPACE FLIGHT CENTER"
     INTERNET            = "ddsrv.hitc.com"
     PORT                = "8080"
    GUIDE_SRV_ADDR      = "http://ddsrv.hitc.com:8080/cgi-bin/ddsrv"
  END_GROUP   = DATA_CENTER

                                        o
                                        o
```

Figure 4-6. Example Excerpt from Archive.odl File Documenting Server Address and HTTP Protocol for Connecting to an ECS Document Data Server

(bold emphasis added) Note: Naming conventions for the DAACs are noted as "[DAAC name]_ECS", to distinguish them from the Version 0 DAACs in the archive.odl file.

4.4.1.2 *Guide Search Results*

Each of the V0 DAAC Guide Servers creates a results "hit" list formatted using HTML, and sends the HTML page over the WAIS connection, independently of the other V0 DAAC Guide Servers. The ECS client does not integrate the results of the multiple V0 DAAC Guide Servers into a merged list. Each document "hit" is displayed as a hyperlink in the guide results window in the ECS client. When the user selects a guide document for viewing, the request is sent to the appropriate V0 Guide Server(s) as an HTTP message using the HTTP protocol. The HTTP message contains the path name of the document and the server address which stores the document. The document is returned over the HTTP connection to the B0SOT or to the JEST client.

4.4.2 Guide Search Requests from V0 to the ECS

The V0 Client sends search requests for documents as WAIS formatted queries over HTTP protocol directly to the ECS Document Data Servers.

Guide searches of the ECS Document Data Server will not be supported in Release B.0. If a Guide Search Request is sent to an ECS DAAC before Release B.1, no guide server entry will be found in the archive.odl file and the V0 IMS Client will give the user a message that there is no guide server at that DAAC.

4.4.2.1 Guide Search Requests

The valids information in the V0 Client determines which guide servers the guide query is sent to. A separate search message is created for each destination server. The fields and values used in the guide search queries are an encoded version of those used in search and order fields and values used for the Directory and Inventory search functions in the ECS Clients. There is a one-to-one correspondence between these which is automatically handled by each ECS Document Data Server. The HTTP servers at the ECS DAACs receive the guide queries, perform the mapping, and process the search using the Science Data Server Data Base Management System (DBMS) interfaces.

4.4.2.2 Guide Search Results

Each of the ECS Document Data Servers creates a results "hit" list formatted with HTML, and sends the HTML page over the HTTP connection, independently of the other ECS Document Data Servers. The V0 Client does not integrate the results of the multiple ECS Document Data Servers into a merged list. Each document "hit" is displayed as a hyperlink in the guide results window in the V0 Client. When the user selects a guide document for viewing, the request is sent as an HTTP message using the HTTP protocol. This HTTP message contains the path name of the document and the server address which stores the document. The document is returned to the V0 Client, over the HTTP connection.

4.5 Inventory Search Request/Results and Acknowledge

The purpose of the inventory search is to aid a user in searching through the available inventory, locating and retrieving metadata about specific granules of the product(s) of interest, and determining whether any granules should be ordered; and also to allow a user to find datasets if the user chooses not to use a directory or guide search first. The search criteria, specified by the user, include the following searchable attributes: source, sensor, geophysical parameter, dataset name, data center id, campaign, processing level, geographical coordinates (area) and temporal intervals.

An ECS or V0 user requesting services from the other system submits the inventory search request via the ECS or V0 client respectively. The client sends an Inventory Search Request that includes inventory search criteria based on characteristics of the data. The respective data server retrieves the requested granules' metadata, and sends an Inventory Search Results message back to the requesting client. In order to accommodate two-way mapping of terminology between ECS and the V0 system, the ECS V0 Gateway reads the ECS Data Dictionary containing the terminology mapping information. The ECS Data Dictionary is built by a Data Dictionary Administrator using V0 System search parameters, ECS schema and metadata. Specifically, upon receiving a request from either side of the interface, the V0 Gateway performs an ECS-V0 mapping table look-up within the Data Dictionary database, converting the ECS or V0 request into the other system’s terminology. Similarly, when results are returned through the V0 Gateway the V0-ECS mapping service converts the results prior to forwarding them to the requesting client.

4.5.1 Chunking

When the Inventory Results generated from a user query are large, an Inventory Results message can be broken up into "chunks" according to a set of rules. The chunks are composed of basic types of information; Inventory Result Prefix, Dataset group, and Granule Group. Package Information can be integrated into the chunks according to three options as follows:

Option 1. - Adding All Package Groups in front of the First Dataset Group

Option 2. - Adding Relevant Package Groups in front of each Dataset Group

Option 3. - Adding Relevant Package Groups in each Dataset Group

The following example illustrates the structure, guidelines, and options for placing Package Information for chunking:

INVENTORY RESULT PREFIX:

Info: (Message_Id, Data_Center, Status_Code, Status_Code_Comment, Unmapped_Field)

Rule: (Required for each chunk)

Option 1 for Package Information (0 - many per chunk)

Option 2 for Package Information (0 - many per chunk)

DATASET GROUP:

Info: (Metadata within the Dataset group)

Rule: (0 - many; avoid repeating in other chunks)

Option 3 for Package Information (0 - many per chunk)

GRANULE GROUP:

Info: (Metadata within the Granule group)

Rule: (0 - many per chunk)

The current implementation requires that each chunk contain at least Inventory Result Prefix information and Dataset group metadata. When this restriction is removed, then a 0 - or - more option will permit the following combinations of information for chunks:

a. Chunk - Inv Result Prefix + Package Information

b. Chunk - Inv Result Prefix + Package Information + Dataset metadata

c. Chunk - Inv Result Prefix + Dataset metadata + Granules

d. Chunk - Inv Result Prefix + Dataset metadata + Package Information + Granules

Chunking can specify the total number of granules returned in an Inventory Results message. Chunks need not be uniform in size although a past guideline constraint for a granule-per-chunk cap of 51, yielding a chunk size of 64Kbytes, is useful but not mandatory. The number of granules per dataset may or may not fit in one chunk. The average size allowed for a chunk helps decide what combination of information will be fit into it. This is affected by the different options selected for sending package information.

Rules and Guidelines:

Size of each chunk need not be uniform but should be moderate in size (64kbytes).

The size of the package information for granules in a particular Inventory Results message can be either consolidated in the first chunk (Option 1 if it fits within the chunk size cap) or distributed in other chunks according to Options 2 and 3.

Avoid repeating the Dataset metadata and package information once provided in an appropriate chunk.

Granules are added to a dataset group until the granules per chunk limit is reached. Remaining granules can be put into the next chunk(s).

Use the NUMBER_OF_GRANULE_HITS field to store the total granule count for the result message following the last granule of each dataset in the sequence.

An Inventory Result chunk can have several dataset groups, *or* granules from a single dataset can be spread across several Inventory Result chunks.

Do not break ODL messages across groups (any chunk is a complete ODL message form).

Chunks come in a sequence.

Each Result message is expected to have the Message_ID and Monitor group added.

Dataset metadata is included only before the first granule.

A chunk can contain more than one dataset and their granules.

The receiving client sends a separate Acknowledge message upon receipt of each chunk.

The Inventory Search Request, Inventory Search Results and Acknowledge messages are implemented using ODL. Their ODL Normalization Forms are defined in the immediately-following sections.

4.5.2 ODL Normalization Form for Inventory Search Request

INVENTORY_SEARCH group ::==

Note: The ECS does not support searches by GRANULE_ID. If ECS receives GRANULE_ID_REQ in an INVENTORY_SEARCH, it will return

```
            STATUS_CODE 10 (requested function not supported) in the
            INVENTORY_RESULTS.
    MESSAGE_ID
    [AUTHENTICATOR]
    [ECS_AUTHENTICATOR]
    GRANULE_LIMIT
    [BROWSE_ONLY]
    [CAMPAIGN]
    [DATASET_ID]
    [SENSOR_NAME]
    [SOURCE_NAME]
    [START_DATE]
    [STOP_DATE]
    [START_DAY_OF_YEAR]
    [STOP_DAY_OF_YEAR]
    [DAY_NIGHT]
    [PROCESSING_LEVEL]
    [PARAMETER]
    GLOBAL_GRANULES_ONLY|POINT_LOC group|POLYGON_LOC
    group|RANGE_LOC group
    MONITOR group
    VERSION group

POINT_LOC group ::==
    LATITUDE
    LONGITUDE

POLYGON_LOC group ::==
    LATITUDE
    LONGITUDE
    [POLE_INCLUDED]
    MAP_PROJECTION_TYPE
    TANGENT_LATITUDE
    TANGENT_LONGITUDE

RANGE_LOC group ::==
    NORTH_LATITUDE
    SOUTH_LATITUDE
    EAST_LONGITUDE
    WEST_LONGITUDE

MONITOR group ::==
    TX_CLIENT
    [RX_SERVER]
    [TX_SERVER]
    [RX_CLIENT]
```

[SESSION_ID]

VERSION group ::==
PROTOCOL_VERSION
SENDER_VERSION
[IMS_STAFF]

4.5.3 ODL Normalization Form for Inventory Search Results

Note 1: Source, sensor and parameter information can be put either in DATASET or GRANULE groups. See annotations.

Note: 2: In the Release B.0 timeframe the ECS will provide APPROX_COST for Landsat-7 data only. For other products, ECS will return 0 (zero) for APPROX_COST.

INVENTORY_RESULT group ::==
MESSAGE_ID
DATA_CENTER_ID
STATUS_CODE[STATUS_CODE_COMMENT]
MONITOR group
(PACKAGE group)* repeated group
OPTION 1: for use when all package information is sent for the whole inventory result.
OPTION 2: for use when package information is sent in front of each relevant dataset group.
(DATASET group)*
[NUMBER_OF_DATASETS] (present only in the last chunk for an inventory results set)
[UNMAPPED_FIELD]
[VERSION group]

PACKAGE group ::==
DATA_CENTER_ID
DATASET_ID
PACKAGE_ID The PACKAGE_ID in the PACKAGE group gives an arbitrary identifier by which the package is known. Processing and media options for the package are provided in the group. GRANULE groups can list multiple packages in which they are available. For the common case where granules can be ordered in single-granule packages and all such packages have the same processing and media options, a single package group can be provided whose id is “*”. Then each granule that can be ordered this way can be listed as being in PACKAGE_ID “*” (along with possibly other named packages).
COMMENT
[INFO_PROMPT]
NUMBER_OF_GRANULES
NUMBER_OF_OPTIONS
(PROCESSING_OPTION group)+

DATASET group ::==
STATUS_CODE
DATASET_ID
(VALID_ACCOUNTS group)* repeated group
(PACKAGE group)* ::== repeated group
OPTION 3: for use when package information is sent within each relevant dataset group and before the granule group(s).
(GRANULE group)* repeated group
[MD_ENTRY_ID]
[SENSOR_NAME] If all granules of the dataset have the same values for SENSOR_NAME, the value can be specified in the DATASET group and omitted from all of the GRANULE groups.)
[SOURCE_NAME] If all granules of the dataset have the same values for SOURCE_NAME, the value can be specified in the DATASET group and omitted from all of the GRANULE groups.)
[PARAMETER] If all granules of the dataset have the same values for PARAMETER_NAME, the value can be specified in the DATASET group and omitted from all of the GRANULE groups.)
[COMMENT]
[RESTRICTION]
[CAMPAIGN]
[DAY_NIGHT]
[PROCESSING_LEVEL]
[NUMBER_OF_GRANULE_HITS] (omitted from all chunks except the one containing the last granule of the dataset)
[BROWSE_PRODUCT_DESCRIPTION] (the headings should be done in UPPERCASE on lines by themselves in the sequence, i.e. PRIMARY PURPOSE, PRODUCT HISTORY, etc.)

VALID_ACCOUNTS group ::==
[ACCOUNT_NUMBER]
[BALANCE]
[ERROR]

GRANULE group ::==
GRANULE_ID
START_DATE
STOP_DATE

[SENSOR_NAME] If all granules of the dataset have the same values for SENSOR_NAME, the value can be specified in the DATASET group and omitted from all of the GRANULE groups.)
[SOURCE_NAME] If all granules of the dataset have the same values for SOURCE_NAME, the value can be specified in the DATASET group and omitted from all of the GRANULE groups.)

```
    [PARAMETER] If all granules of the dataset have the same values for
        PARAMETER_NAME, the value can be specified in the DATASET group and omitted
        from all of the GRANULE groups.)
    [BROWSE_TYPE]
    [CAMPAIGN]
    [COMMENT]
    [DAY_NIGHT]
    [PROCESSING_LEVEL]
    [PACKAGE_ID] (If omitted or if package information is not provided within the inventory
        results, granule cannot be ordered.)
    GLOBAL_GRANULE|POINT_LOC group|POLYGON_LOC group|RANGE_LOC group

POINT_LOC group ::==
    LATITUDE
    LONGITUDE

POLYGON_LOC group ::==
    LATITUDE
    LONGITUDE
    [POLE_INCLUDED]
    CENTROID_LAT
    CENTROID_LON

RANGE_LOC group ::==
    NORTH_LATITUDE
    SOUTH_LATITUDE
    EAST_LONGITUDE
    WEST_LONGITUDE

PROCESSING_OPTION group ::==
    OPTION_ID
    PACKAGE_SIZENUMBER_OF_MEDIA_TYPE
    (MEDIA_TYPE group)+


MEDIA_TYPE group ::==
    TYPE_ID
    NUMBER_OF_MEDIA_FORMAT
    (MEDIA_FORMAT  group)+


MEDIA_FORMAT group ::==
    FORMAT_ID
    APPROX_COST


MONITOR group ::==
    TX_CLIENT
```

RX_SERVER
TX_SERVER
[RX_CLIENT]
[SESSION_ID]

VERSION group ::==Optional group
PROTOCOL_VERSION
SENDER_VERSION
[IMS_STAFF]

4.5.4 ODL Normalization Form for Acknowledge

ACKNOWLEDGE group ::==
[MESSAGE_ID]
MONITOR group
VERSION group

MONITOR group ::==
TX_CLIENT
[RX_SERVER]
[TX_SERVER]
[RX_CLIENT]
[SESSION_ID]

VERSION group ::==
PROTOCOL_VERSION
SENDER_VERSION
[IMS_STAFF]

4.6 Browse Request/Results

The purpose of the Browse service is to allow the user to request and receive representative images for FTP-copying or viewing and for analysis prior to deciding on specific full-resolution products to order. Two browse service modes are available to a user: FTP browse and integrated browse. The FTP Browse service allows the user to FTP-copy the browse product on to the user's system. The Integrated Browse service allows the user to view the browse product through the user's client. For some data types browse modes are not available. [It is important to note that with the V0 Client or B0SOT the user must do an inventory search and view the inventory data before ordering browse data.]

FTP Browse Request: After the data server receives an FTP browse request (BROWSE_TYPE = FTP_Only), it transmits the FTP Browse Results, to the requesting client, and copies the browse product onto the FTP pull staging area.

The user is notified via email that the browse product is ready to FTP. The email notification provides the access information for the FTP including the following: (1) FTP account, (2) IP address of the FTP host, (3) directory and file name for each of the requested browse products, (4) pick up expiration date of staged files. The FTP browse product remains on the FTP site for an operator-tunable time interval, giving the user time to copy it onto the user's system.

Integrated Browse Request: In response to an integrated browse request (BROWSE_TYPE = Y) sent by a client to a data server, the data server sends back to the requesting client, the integrated browse results message, followed by the browse image, which is then displayed to the user.

Viewing: All V0 browse images (FTP and integrated) are provided in the National Center for Supercomputing Applications (NCSA) Hierarchical Data Format (HDF). ECS browse images during the Release B.0 timeframe are in HDF-EOS format, which is simply an extension of HDF. For Release B.0, the HDF-EOS version is 2.0. It is based on HDF Version 4.1, which is backward compatible with earlier versions. The HDF and HDF-EOS versions for other ECS releases are to be determined.

Browse images can be viewed with EOSView. EOSView is an interactive software tool which aids a user in the selection, verification, presentation, and analysis of browse data files written in HDF or HDF-EOS format. Specifically, EOSView is designed to help the user to interactively visualize HDF or HDF-EOS browse data files during the selection of data, to verify that the data received is the data desired, and to get data which resides in some of the more common visualization and analysis systems, such as interactive data language (IDL).

EOSView can be run as a standalone application and used to view FTP browse data on the user's workstation.

EOSView can be run concurrently with the ECS B0SOT. The B0SOT, on its own, can display V0 image layers of HDF or HDF-EOS browse data files. It can also save the file to a user-selectable directory for viewing by EOSView or other viewers.

At Release B.1, JEST will have integrated browse capability for HDF files below approximately 1 MB. JEST will allow the user to save the browse file to the user's local workstation for later viewing by EOSView. The integrated browse capability in JEST will allow the user to mark a granule for ordering.

The V0 Client can display the image layers of ECS browse data files written in HDF-EOS format. This helps the V0 user to visualize ECS browse images during the selection of data and to verify that the data received is the data desired. It is important to point out that the V0 Client is not capable of displaying text, table or movie loop documents. The V0 Client can also save a browse file in a user-selectable directory for viewing with other viewers such as EOSView. The Browse Request/Results messages are implemented using ODL. Their ODL Normalization Forms are defined in the immediately-following sections.

4.6.1 ODL Normalization Form for Browse Request

```
BROWSE_REQUEST group ::==
    MESSAGE_ID
    [AUTHENTICATOR]
    [ECS_AUTHENTICATOR]
    DATA_CENTER_ID
    [USER_AFFILIATION group]
    BROWSE_TYPE
    BROWSE_GRANULES group
    CONTACT_ADDRESS group
    MONITOR group
    VERSION group

BROWSE_GRANULES ::==
    DATASET_ID
    GRANULE_ID

CONTACT_ADDRESS group ::==
    [TITLE]
    LAST_NAME
    FIRST_NAME
    [MIDDLE_INITIAL]
    [ORGANIZATION]
    ADDRESS
    CITY
    [STATE]
    [ZIP]
    COUNTRY
    PHONE
    [FAX]
    EMAIL

MONITOR group ::==
    TX_CLIENT
    [RX_SERVER]
    [TX_SERVER]
    [RX_CLIENT]
    [SESSION_ID]

VERSION group ::==
    PROTOCOL_VERSION
    SENDER_VERSION
    [IMS_STAFF]

USER_AFFILIATION group ::==
    CATEGORY
    TYPE
```

4.6.2 ODL Normalization Form for FTP Browse Results

```
FTP_BROWSE_RESULT group ::==
    MESSAGE_ID
    DATA_CENTER_ID
    STATUS_CODE
    [STATUS_CODE_COMMENT]
    TOTAL_FILE_SIZE
    (DAAC_CONTACT_ADDRESS group)+
    MONITOR group
    VERSION group

DAAC_CONTACT_ADDRESS group ::==
    CONTACT_NAME
    ORGANIZATION
    [ADDRESS]
    CITY
    [STATE]
    [ZIP]
    COUNTRY
    PHONE
    [FAX]
    [EMAIL]

MONITOR group ::==
    TX_CLIENT
    RX_SERVER
    TX_SERVER
    [RX_CLIENT]
    [SESSION_ID]

VERSION group ::==
    PROTOCOL_VERSION
    SENDER_VERSION
    [IMS_STAFF]
```

4.6.3 ODL Normalization Form for Integrated Browse Results

```
INTEGRATED_BROWSE_RESULT ::==
    MESSAGE_ID
    DATA_CENTER_ID
    STATUS_CODE
    [STATUS_CODE_COMMENT]
    IMAGE group
    MONITOR group
    [VERSION group]

IMAGE group ::==
    DATASET_ID
    GRANULE_ID
    IMAGE_ID
    IMAGE_SIZE

MONITOR group ::==
    TX_CLIENT
    RX_SERVER
    TX_SERVER
    [RX_CLIENT]
    [SESSION_ID]

VERSION group ::==Optional group
    PROTOCOL_VERSION
    SENDER_VERSION
    [IMS_STAFF]
```

The INTEGRATED_BROWSE_RESULT message is followed by the browse file itself transferred as a binary stream of IMAGE_SIZE bytes.

4.7 Product Request/Result

The Product Request allows the V0 or ECS user to order data products from the other system through the user's client. After the user has successfully searched, located, and viewed the inventory data for the datasets and selected the granules desired, and possibly viewed certain representative browse images, the user may, but is not required to, submit a product request.
[It is important to note that with the V0 Client or B0SOT the user must do an inventory search and view the inventory data before ordering data.]

The Product Result is sent from the data server to the requesting client. It confirms receipt of the Product Request and provides contact information for further inquiries. The actual product is distributed via hard media or to an appropriate FTP site.

4.7.1 ODL Normalization Form for Product Request

Note: 1: In the Release B.0 timeframe the ECS will provide EST_COST for Landsat-7 data only. For other products, ECS will return 0 (zero) for EST_COST.

```
PRODUCT_REQUEST group ::==
    MESSAGE_ID
    REQUEST_ID
    DATA_CENTER_ID
    [AUTHENTICATOR]
    [ECS_AUTHENTICATOR]
    [INITIAL_USER_KEY]
    USER_AFFILIATION group
    CONTACT_ADDRESS group
    [SHIPPING_ADDRESS] group
    [BILLING_ADDRESS] group
    (LINE_ITEM group)+          repeated group
    MONITOR group
    VERSION group

USER_AFFILIATION group ::==
    CATEGORY
    TYPE

CONTACT_ADDRESS group ::==
    [TITLE]
    LAST_NAME
    FIRST_NAME
    [MIDDLE_INITIAL]
    USER_AFFILIATION]
    [ORGANIZATION]
    ADDRESS
    CITY
    [STATE]
    [ZIP]
    COUNTRY
    PHONE
    [FAX]
    EMAIL

SHIPPING_ADDRESS group ::==  Optional group
    [TITLE]
    LAST_NAME
    FIRST_NAME
    [MIDDLE_INITIAL]
    [ORGANIZATION]
    [ADDRESS]
    CITY
    [STATE]
    [ZIP]
    COUNTRY
    PHONE
    [FAX]
    [EMAIL]

BILLING_ADDRESS group ::== Optional group
    [TITLE]
    LAST_NAME
    FIRST_NAME
    [MIDDLE_INITIAL]
    [ORGANIZATION]
    [ADDRESS]
    CITY
    [STATE]
    [ZIP]
    COUNTRY
    PHONE
    [FAX]
    [EMAIL]

LINE_ITEM group ::==
    DATASET_ID
    [PACKAGE_ID]
    PROCESSING_OPTIONS
    MEDIA_TYPE
    MEDIA_FORMAT
    [ADDITIONAL_INFO]
    [BILLING_ID]
    [EST_COST]

MONITOR group ::==
    TX_CLIENT
    [RX_SERVER]
    [TX_SERVER]
    [RX_CLIENT]
    [SESSION_ID]

VERSION group ::==
    PROTOCOL_VERSION
    SENDER_VERSION
    [IMS_STAFF]
```

4.7.2 ODL Normalization Form for Product Results

```
PRODUCT_RESULT group ::==
    MESSAGE_ID
    DATA_CENTER_ID
    STATUS_CODE
    [STATUS_CODE_COMMENT]
    (DAAC_CONTACT_ADDRESS group)+       (repeatable mostly to support "DAACs" that
                        are consortia of multiple archives in the international community)
    MONITOR  group
    [VERSION group]
DAAC_CONTACT_ADDRESS group ::==
    CONTACT_NAME
    ORGANIZATION
    [ADDRESS]
    CITY
    [STATE]
    [ZIP]
    COUNTRY
    PHONE
    [FAX]
    [EMAIL]
MONITOR group ::==
    TX_CLIENT
    RX_SERVER
    TX_SERVER
    [RX_CLIENT]
    [SESSION_ID]
VERSION group ::==Optional group
    PROTOCOL_VERSION
    SENDER_VERSION
    [IMS_STAFF]
```

4.8 Quit

If problems necessitate premature termination of the process, a quit message is transmitted between the Server and the client. (Note: As appropriate, read V0 Gateway for client or server as indicated in Figure 4-2, 4-3 or 4-4.) Specifically, the client sends a quit message to the Server if the user presses the "abort" button on the screen. On the other hand, the quit message is sent by the Server to the client if an error condition terminates the response.

Quit messages are also used to synchronize the client with the Server following the last chunk in an inventory result. The Server sends a QUIT with a STATUS_CODE of 1 to the client and the connection is dropped.

4.8.1 ODL Normalization Form for Quit

```
QUIT group ::==
    [AUTHENTICATOR]
    [ECS_AUTHENTICATOR]
    MESSAGE_ID
    [DATA_CENTER_ID]
    STATUS_CODE
    [STATUS_CODE_COMMENT]

    MONITOR group
    VERSION group

MONITOR group ::==
    TX_CLIENT
    [RX_SERVER]
    [TX_SERVER]
    [RX_CLIENT]

    [SESSION_ID]

VERSION group ::==
    PROTOCOL_VERSION
    SENDER_VERSION
    [IMS_STAFF]
```

5. Dependent Valids

5.1 General

To achieve Level 3 interoperability, dependent valids information must be made available to support V0 directory searches by ECS users and vice versa. Section 5.2 defines the format that each DAAC uses to provide valids information to the V0-ECS Science Team and, at Release B.1, to the ECS. Section 5.3 specifies the format into which, under the auspices of the V0-ECS Science Team, those files are integrated for provision to the V0 IMS Client and B0SOT.

5.2 Valids Input from V0 DAACs

Each time a V0 DAAC updates its valids, it produces an updated ODL Valids File in accordance with the form defined in Section 5.2.1. This file is used to supply valids to ECS at both Release B.0 (B0SOT client) and Release B.1 (ECS Advertising Service and Data Dictionary).

To support the V0 and B0SOT clients, the V0 DAAC places the Valids File in an ftp directory for pickup and sends an email notification to the V0-ECS Science Team, which picks up the files to use in preparing valids support files (see Section 5.3).

At Release B.1, each V0 DAAC also provides the same Valids File to the ECS DAAC Operator in accordance with procedures to be agreed on. The ECS Operator uses the IMPORT function of the Data Dictionary Maintenance tool to import the valids information. ECS extracts the V0 valids and maps them to the ECS valids structure to populate the ECS Advertising Service and Data Dictionary, which are used by the JEST client.

5.2.1 ODL Normalization Form, V0 DAAC Dependent Valids Submittal

For ODL conventions followed here, see Figure 4-1. For more detailed information, see the keyword definitions in Appendix A.2 of this document and also “A Format for Valids and Keyword Definitions Using ODL” by Patrick M. Ryan, Hughes STX Corporation, Revision 1.5, October 12, 1994.

Note: Although some of the keywords employed here also appear in the V0 message protocol, their definitions may differ when they are used in the VALIDS message form. Therefore, be sure to use Appendix A.2 as the primary reference for descriptions. If the keyword description is not in Appendix A.2, refer to Appendix A.1.

VALIDS group::==

DATA_CENTER_ID

(DATASET group)+

(The VALIDS group itself can be omitted and just contents of the group, i.e., the DATA_CENTER_ID followed by one or more DATASET, can be included in the valids file submitted.)

DATASET group::==

BROWSE group

[CAMPAIGN]

DATASET_COVERAGE group

DATASET_ID

[DATE_AVAILABLE]

[DAY_NIGHT_FLAG]

(DEPENDENCY group)*

[EXTENDED_CRITERIA_AVAIL]

FTP_PRODUCT_AVAILABLE

GRANULE_COVERAGE group

MD_ENTRY_ID

[PARAMETER]

PROCESSING_LEVEL

[SENSOR]

[SOURCE]

BROWSE group::==

FTP

INTEGRATED

DATASET_COVERAGE group::==

SPATIAL

TEMPORAL

DEPENDENCY group::==

[PARAMETER]

[SENSOR]

[SOURCE]

GRANULE_COVERAGE group::==

SPATIAL

TEMPORAL

5.3 Science Team Output to V0 and B0SOT Clients

A single master Valids File combining valids information from all of the V0 DAACs is maintained under the auspices of a V0-ECS Science team. The V0-ECS Science Team reviews and integrates the DAAC valids files into a master Valids File and Valids Support Files. A TAR file containing the support file package is placed on a designated ftp server for pickup by the V0 IMS and B0SOT clients and the V0 Gateway. Each time a V0 or B0SOT client is initialized, it runs a script (called AUTOXFER) to download the current valids information. The format of the combined Valids File is defined in Section 5.3.1 and that of the Valids Support Files, in Section 5.3.2.

5.3.1 Valids File Format

The Valids File format for each line is as follows.

Number, 1 space, KEYWORD=, VALID STRING, {ALIAS1} ... {ALIASn}

where

- Number represents the valid index number (i.e., bit position of valid)
- KEYWORD represents the name of the field (i.e., the field type)---currently, there are 7 field types, including the following:
  - DATA_CENTER_ID
  - DATASET_ID
  - SOURCE
  - SENSOR
  - PARAMETER
  - CAMPAIGN
  - PROCESSING_LEVEL
- VALID STRING represents the actual name of the valid (e.g., GSFC, MODIS, TRMM, etc.). There shall be no space after the "=" symbol and before the VALID STRING---i.e., KEYWORD=VALID STRING (for example, SENSOR=MODIS).

- ALIAS names are alternate names for the VALID STRING. ALIAS names are always enclosed in braces {} and are limited to 80 characters each (not including the braces). If there is no ALIAS the field will be empty.
- The lines of the Valids File are sorted in numerical order by valid index Number.

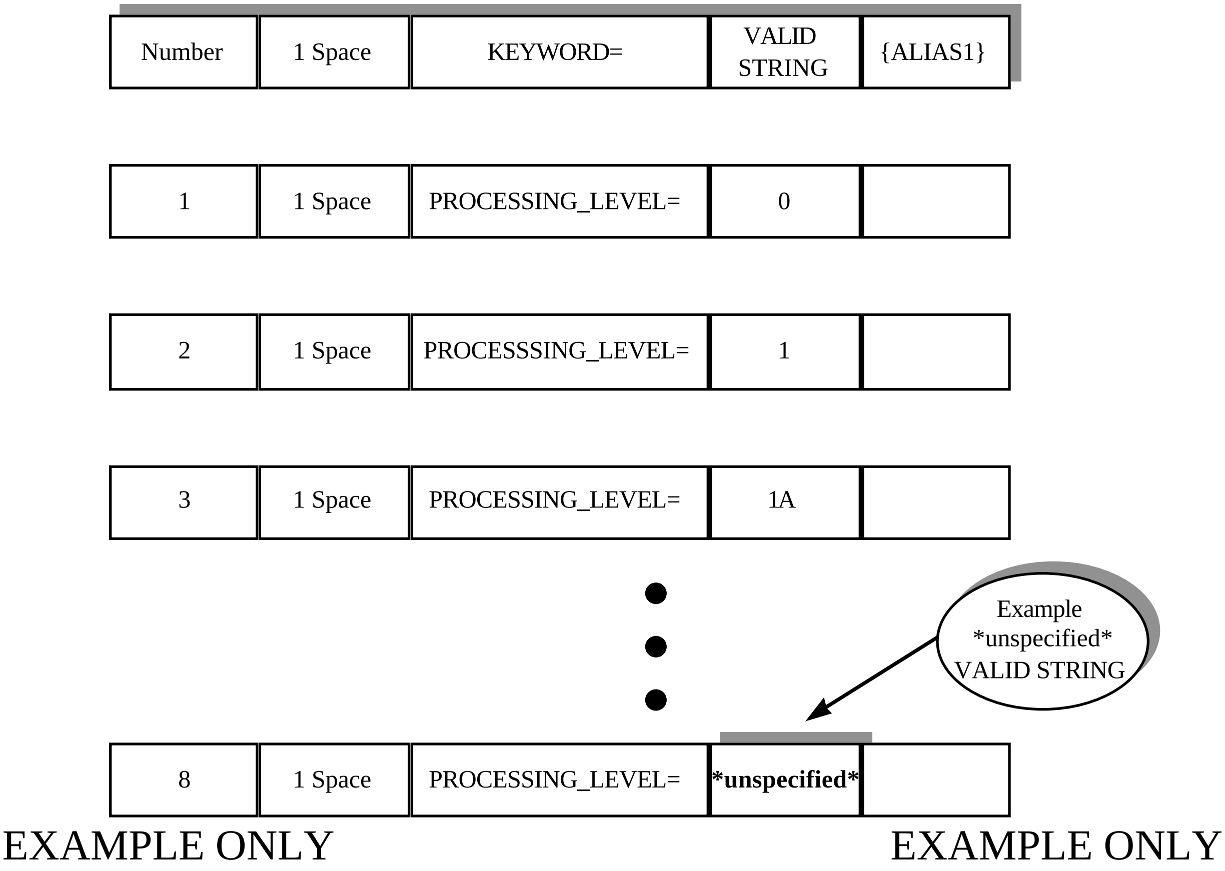
An example of the format of the Valids File is depicted in Figure 5-1.

For fields other than DATA_CENTER_ID or DATASET_ID, *unspecified* is a VALID STRING that may be used to indicated an undefined value. An example *unspecified* VALID STRING is depicted in Figure 5-2.

Number	1 Space	KEYWORD=	VALID STRING	{ALIAS1}	{ALIAS2}
5	1 Space	SENSOR=	SAGE-II	{SAGE-2}	{SAGE_II}

EXAMPLE ONLY **EXAMPLE ONLY**

Figure 5-1. Example Format of Valids Master File



Number	1 Space	KEYWORD=	VALID STRING	{ALIAS1}
1	1 Space	PROCESSING_LEVEL=	0	
2	1 Space	PROCESSSING_LEVEL=	1	
3	1 Space	PROCESSING_LEVEL=	1A	
		● ● ●		
8	1 Space	PROCESSING_LEVEL=	***unspecified***	

Figure 5-2. Example *unspecified* VALID STRING

5.3.2 Valids Support Files

The Valids Support TAR File consists of the following files:

a. Bitmap header

b. Bitmap file (including bitmap header)

c. Field bitmap file (including bitmap header)

d. Filter bitmap file (including bitmap header)

e. Filter field bit mask file (including bitmap header)

f. Valid string list

g. Filter string list

a. Bitmap Header

The bitmap header will be a character string of 80 characters where the last character is a newline character ("\n"). The bitmap header will include, at a minimum, the following items of information:

a. a header version number

b. one (1) space

c. a serial number

d. one (1) space

e. date

f. one (1) space

g. number of rows in the bitmap

h. one (1) space

i. number of columns of active bits

j. one (1) space

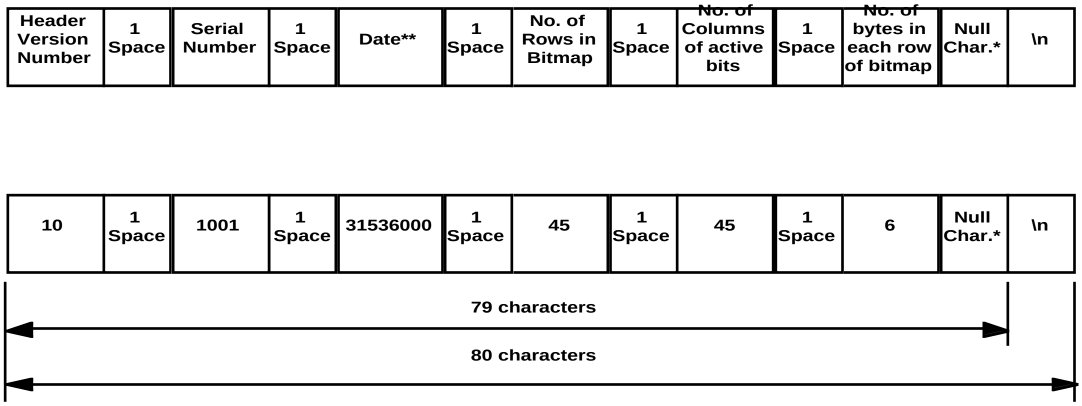
k. number of bytes in each row of the bitmap

l. NULL characters to pad to 79 characters

m. the newline character ("\n")

Here, "date" represents the time when the New Valids Ingest Software wrote the file (in seconds since 00:00:00 UTC, January 1, 1970). This same header format will be used for the Bitmap File, Field Bit Mask File, Filter Bitmap File, and Filter Field Bit Mask File discussed, respectively, in Sections 5.3.2b through 5.3.2e. An example bitmap header is depicted in Figure 5-3.

EXAMPLE ONLY **EXAMPLE ONLY**



Header Version Number	1 Space	Serial Number	1 Space	Date**	1 Space	No. of Rows in Bitmap	1 Space	No. of Columns of active bits	1 Space	No. of bytes in each row of bitmap	Null Char.*	\n
10	1 Space	1001	1 Space	31536000	1 Space	45	1 Space	45	1 Space	6	Null Char.*	\n

* NULL characters pad to 79 characters
** Represents time in seconds since 00:00:00 UTC, Jan 1, 1970

EXAMPLE ONLY **EXAMPLE ONLY**

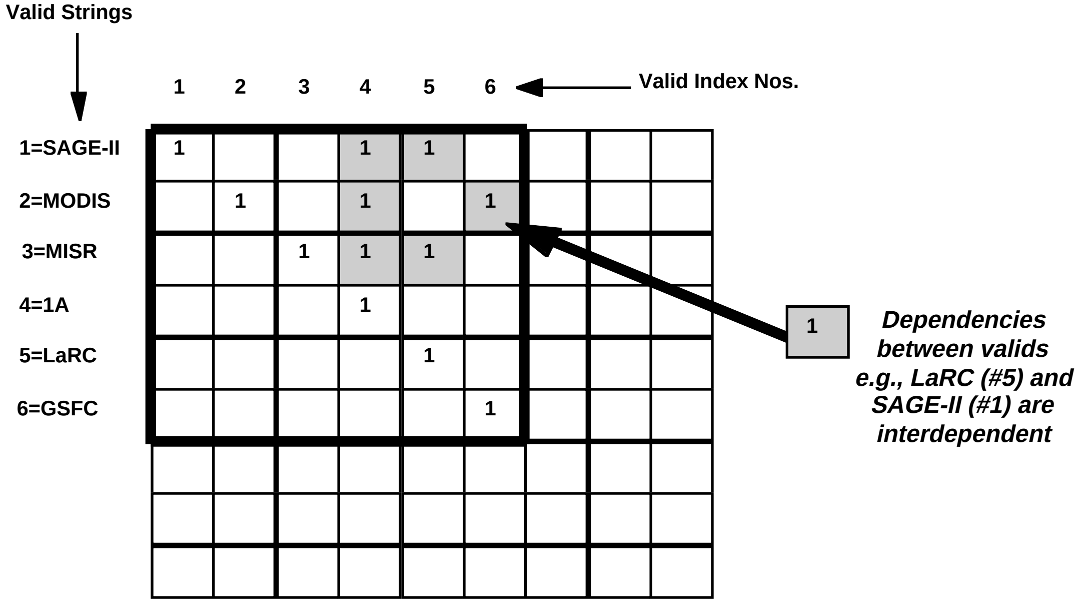
Figure 5-3. Example Bitmap Header

b. Bitmap File

The bitmap file contains a bitmap header and the bitmap data. Within the header, the "Number of Rows in Bitmap" and the "Number of Columns of Active Bits" are the same and they represent the total number of valids. The "Number of Bytes in Each Row of Bitmap" represents the number of bytes needed to store the number of columns of bits. The bitmap represents the dependencies between the valids. Each row represents a valid and its index is the same as in the master list of valid strings. A bit is "on" when the valid bit column number is compatible with the current valid row. The order of bytes within the bitmap is: a row of bytes followed by the next row of bytes. The order of bits within a byte is the "zeroeth" bit starts on the left. Some bits in the last byte of the row may not be used if the number of bits is not divisible by the number of bits per byte. If the bits are not used they shall be set to zero. An example bitmap file is depicted in Figure 5-4.

EXAMPLE ONLY **EXAMPLE ONLY**

Header Version Number	1 Space	Serial Number	1 Space	Date**	1 Space	No. of Rows in Bitmap	1 Space	No. of Columns of active bits	1 Space	No. of bytes in each row of bitmap	Null Char.*	\n
10	1 Space	1001	1 Space	31536000	1 Space	6	1 Space	6	1 Space	1	Null Char.*	\n


Valid Strings
Valid Index Nos.
1
Dependencies between valids e.g., LaRC (#5) and SAGE-II (#1) are interdependent

	1	2	3	4	5	6
1=SAGE-II	1			1	1	
2=MODIS		1		1		1
3=MISR			1	1	1	
4=1A				1		
5=LaRC					1	
6=GSFC						1

EXAMPLE ONLY **EXAMPLE ONLY**

Figure 5-4. Example Bitmap File

c. Field Bit Mask File

The field bit mask file contains a bitmap header and the field bit mask data. In this case within the header, the number of rows is set to the number of field types. Currently, the number of field types is seven (7). The number of columns represents the total number of valids. The number of bytes represents the number of bytes needed to store the number of columns of bits. The field bit mask indicates which field each valid represents. Each row of the field bit mask represents one of the 7 field types. A bit is "on" for the valid bit column number, when the valid belongs to the field represented by the row. The order of bytes within the field bit mask is: a row of bytes followed by the next row of bytes. The order of the rows is as follows: DATA_CENTER_ID, DATASET_ID, SOURCE, SENSOR, PARAMETER, CAMPAIGN, and PROCESSING_LEVEL. The order of bits within a byte is the "zeroeth" bit starts on the left. Some bits in the last byte of the row may not be used if the number of bits is not divisible by the number of bits per byte. If the bits are not used they shall be set to zero. An example field bit mask file is depicted in Figure 5-5.

EXAMPLE ONLY **EXAMPLE ONLY**

Header Version Number	1 Space	Serial Number	1 Space	Date**	1 Space	No. of Rows in Bitmap	1 Space	No. of Columns of active bits	1 Space	No. of bytes in each row of bitmap	Null Char.*	\n
10	1 Space	1001	1 Space	31536000	1 Space	7	1 Space	11	1 Space	2	Null Char.*	\n

Field Types

Valid Index Nos.

	1	2	3	4	5	6	7	8	9	10	11
DATA_CENTER_ID	1										1
DATASET_ID			1					1			
SOURCE					1						
SENSOR						1					
PARAMETER				1			1				
CAMPAIGN									1		
PROCESSING_LEVEL		1								1	

EXAMPLE ONLY **EXAMPLE ONLY**

Figure 5-5. Example Field Bit Mask File

d. Filter Bitmap File

The filter bitmap file contains a bitmap header and the filter bitmap data. The number of rows represents the number of filters. The number of bit columns represents the total number of valids. The number of bytes represents the number of bytes required to store the total number of columns of bits. Each row represents a filter. The bits are "on" for those columns that represent the valids for that filter. The order of bytes within the bitmap is: a row of bytes followed by the next row of bytes. The order of bits within a byte is the "zeroeth" bit starts on the left. Some bits in the last byte of the row may not be used if the number of bits is not divisible by the number of bits per byte. If the bits are not used they shall be set to zero. An example filter bitmap file is depicted in Figure 5-6.

EXAMPLE ONLY **EXAMPLE ONLY**

Header Version Number	1 Space	Serial Number	1 Space	Date**	1 Space	No. of Rows in Bitmap	1 Space	No. of Columns of active bits	1 Space	No. of bytes in each row of bitmap	Null Char.*	\n
10	1 Space	1001	1 Space	31536000	1 Space	6	1 Space	12	1 Space	2	Null Char.*	\n

Filters

Valid Index Nos.

Filters	1	2	3	4	5	6	7	8	9	10	11	12		
DATA_CENTER_ID=DAAC 1,2	1	1												
DATASET_ID=CLOUDS 3,4			1	1										
SOURCE=EDOS 5,6					1	1								
SENSOR=RADIOMETER 7,8							1	1						
PARAMETER=AIR_QUALITY 9.10									1	1				
CAMPAIGN=RAD_BUDGET 11,12											1	1		
PROCESSING_LEVEL=MAPS 13,14													1	1

EXAMPLE ONLY **EXAMPLE ONLY**

Figure 5-6. Example Filter Bitmap File

e. Filter Field Bit Mask File

The filter field bit mask file contains a bitmap header and the filter field bit mask data. In this case within the header, the number of rows is set to the number of field types. Currently, the number of field types is seven (7). The number of columns represents the total number of filters. The number of bytes represents the number of bytes needed to store the number of columns of bits. The field bit mask indicates which field each filter represents. Each row of the field bit mask represents one of the 7 field types. A bit is "on" for the filter bit column number, when the filter belongs to the field represented by the row. The order of bytes within the field bit mask is: a row of bytes followed by the next row of bytes. The order of the rows is: DATA_CENTER_ID, DATASET_ID, SOURCE, SENSOR, PARAMETER, CAMPAIGN, and PROCESSING_LEVEL. The order of bits within a byte is the "zeroeth" bit starts on the left. Some bits in the last byte of the row may not be used if the number of bits is not divisible by the number of bits per byte. If the bits are not used they shall be set to zero. An example filter field bit mask file is depicted in Figure 5-7.

EXAMPLE ONLY

EXAMPLE ONLY

Header Version Number	1 Space	Serial Number	1 Space	Date**	1 Space	No. of Rows in Bitmap	1 Space	No. of Columns of active bits	1 Space	No. of bytes in each row of bitmap	Null Char.*	\n
10	1 Space	1001	1 Space	31536000	1 Space	7	1 Space	11	1 Space	2	Null Char.*	\n

Field Types

Filters

	1	2	3	4	5	6	7	8	9	10	11
DATA_CENTER_ID	1	1									
DATASET_ID			1	1							
SOURCE					1	1					
SENSOR							1				
PARAMETER								1			1
CAMPAIGN									1		
PROCESSING_LEVEL										1	

EXAMPLE ONLY

EXAMPLE ONLY

Figure 5-7. Example Filter Field Bit Mask File

f. Valid String List

The Valid String List begins with a string list header which contains the following:

a. header version number (character string)
b. one (1) space
c. serial number of the string list (character string)
d. one (1) space
e. UNIX date in seconds from 1/1/70 00:00:00 UTC (character string)
f. a newline character (\n)

The Valid String List contains the list of strings of all valid types. There shall be one valid string (with its aliases) per line. Each string is terminated with a newline character ("\n"). Missing valids shall be indicated by a single newline character. It shall be guaranteed that this list is sorted by Valids Index Number. An example Valid String List is depicted in Figure 5-8.

Header Version No.	1 Space	Serial No. of String List	1 Space	UNIX date	\n
10	1 Space	1001	1 Space	31200095	\n

VALID STRING	{ALIAS1}
GSFC	{Goddard}
EP/TOMS Data	{TOMS Ozone}
Earth Probe	{EP}
⋮	
1A	N/A

EXAMPLE ONLY **EXAMPLE ONLY**

Figure 5-8. Example Valid String List

g. Filter String List

The Filter String List begins with a string list header. The string list header contains the following:

a. header version number in a character string

b. one (1) space

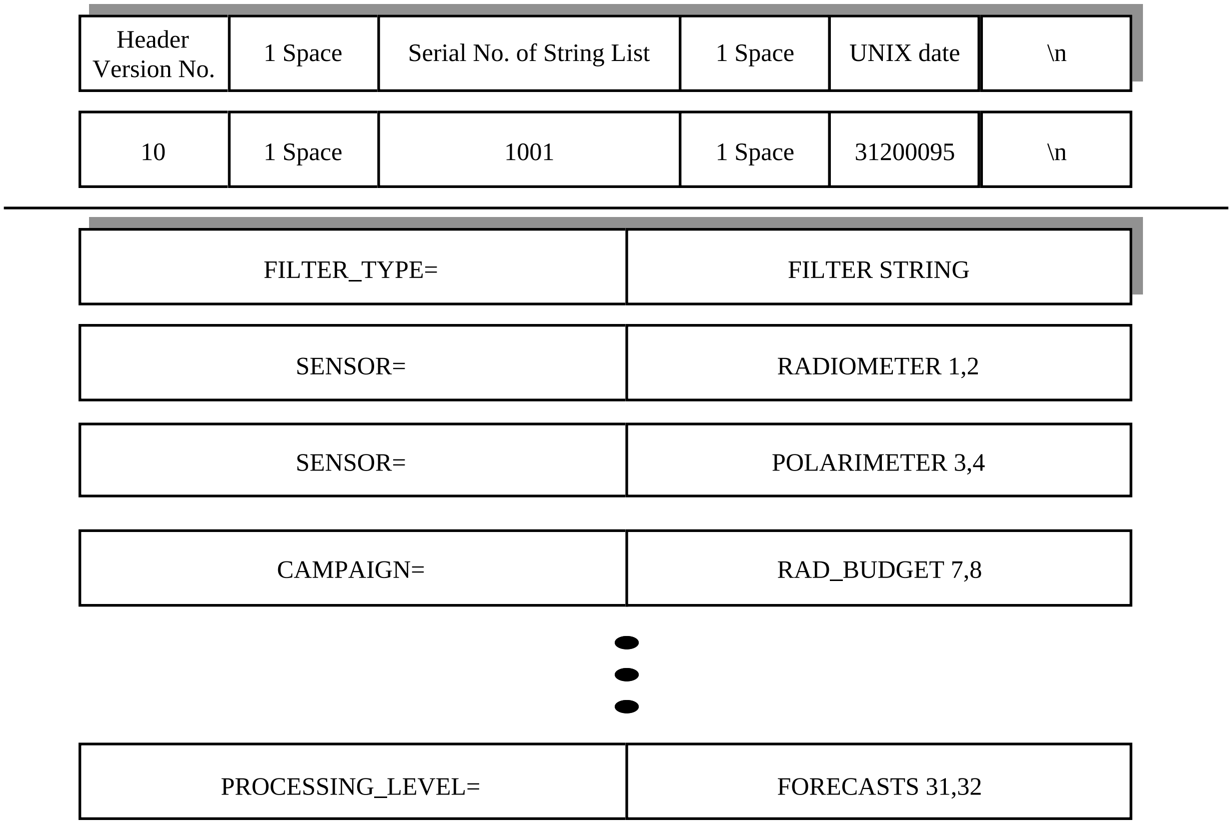
c. serial number of the string list in a character string

d. one (1) space

e. UNIX date in seconds from 1/1/70 00:00:00 UTC in a character string

f. a newline character (\n)

The Filter String List contains the list of strings consisting of all filter names. There shall be one filter string per line. Each string is terminated with a newline character ("\n"). Missing filters shall be indicated by a single newline character. This list is sorted by filter index order. An example filter string list is depicted in Figure 5-9.



Header Version No.	1 Space	Serial No. of String List	1 Space	UNIX date	\n
10	1 Space	1001	1 Space	31200095	\n

FILTER_TYPE=	FILTER STRING
SENSOR=	RADIOMETER 1,2
SENSOR=	POLARIMETER 3,4
CAMPAIGN=	RAD_BUDGET 7,8
PROCESSING_LEVEL=	FORECASTS 31,32

EXAMPLE ONLY **EXAMPLE ONLY**

Figure 5-9. Example Filter String List

Appendix A. ODL Message Keywords (Objects)

This Appendix identifies and defines each of the ODL Message keywords corresponding to the ODL descriptions provided in Sections 4 and 5 of this document. Each keyword is defined, as applicable, in terms of synopsis (short English-Language description of the keyword), parent groups, children, ODL type [e.g., integer, real, date, string, aggregate (i.e., the keyword object contains children), symbol, sequence string (i.e., 0 or more strings entered on separate lines), and character string.], maximum (value) length, and possible values. If no possible values are specified, then any possible value for the stated ODL type is legal. For example, an ACCOUNT_NUMBER may be any string up to 80 characters.

Section A.1 lists keywords used in the V0 interoperability messages described in Section 4 of this document. Section A.2 lists additional keywords used only for submitting dependent valids information as defined in Section 5.

A.1 Keywords Used in V0 Message Protocol

The ODL keywords described in this section are included to support V0-ECS interoperability. They are derived from the "Messages and Development Data Dictionary - V0 and Release A Message Passing Protocol Specification," September 1995. That document is still the baseline for V0-ECS interoperability in ECS Release B. Corrections published in Messages and Development Data Dictionary, IMSV0-PD-SD-002 v2.1, September 1997, have been incorporated and keywords no longer supported by V0 have been removed. However, no other changes published in the May 1997 edition are supported for V0-ECS interoperability.

Keyword: ACCOUNT_NUMBER
Synopsis: Account identifier provided by a DAAC.
Parent Group(s): VALID_ACCOUNTS

ODL Type: String
Maximum Length: 80

Keyword: ACKNOWLEDGE

Synopsis: Message sent to acknowledge receipt by client of inventory search chunk or inventory browse file.

Parent Group(s):

Child Groups: [MESSAGE_ID], MONITOR, VERSION

ODL Type: Aggregate

Keyword: ADDITIONAL_INFO
Synopsis: User supplied information about order, applied to each package (line item).
Parent Group(s): LINE_ITEM

ODL Type: String
Maximum Length: 80

Keyword: ADDRESS
Synopsis: Mailing (street) address for parent group, up to three lines. Should be no more than 3 strings in the sequence.
Parent Group(s): BILLING_ADDRESS, CONTACT_ADDRESS, SHIPPING_ADDRESS, DAAC_CONTACT_ADDRESS
ODL Type: Sequence String
Field length: 32 x 3 (96)

Keyword: APPROX_COST
Synopsis: Estimated cost for the selected data package.
Parent Group(s): MEDIA_FORMAT
ODL Type: Real
Maximum Length: 16
Possible value(s): 0.0 to 9999999999999.99

Keyword: AUTHENTICATON_KEY
Synopsis: Password provided by user for accessing restricted data. Identified on profile screen as “Restricted Data Access Key.”
ODL Type: String
Maximum Length: 16

Keyword: AUTHENTICATOR
Synopsis: Encrypted value from authentication key, last name, first name. Passed with every request (if authentication key is not blank). If AUTHENTICATON_KEY is null, so will this be. Encryption is blind to case and non-alphanumeric characters.
Parent Group(s): BROWSE_REQUEST, PRODUCT_REQUEST, INVENTORY_SEARCH, DIRECTORY_SEARCH
ODL Type: String
Maximum Length: 16

Keyword: BALANCE
Synopsis: Dollar amount remaining for a particular account.
Parent Group(s): VALID_ACCOUNTS
ODL Type: Real
Maximum Length: 16

Keyword: BILLING_ADDRESS
Synopsis: Billing address for data order.
Parent Group(s): PRODUCT_REQUEST
Child Group(s): [ADDRESS], CITY, [EMAIL], [FAX], FIRST_NAME, [MIDDLE_INITIAL], LAST_NAME, PHONE, [STATE], COUNTRY, [ZIP], [TITLE], [ORGANIZATION]
ODL Type: Aggregate

Keyword: BILLING_ID
Synopsis: Account number that the user enters or selects from the ACCOUNT_NUMBERs in the VALID_ACCOUNTS
Parent Group(s): LINE_ITEM
ODL Type: String
Maximum Length: 80

Keyword: BROWSE_GRANULES
Synopsis: Granules of browse request. In spite of name, a single granule is identified for the browse request. Each file, whether ftp or integrated, is specified in its own request. (The capability to request multiple ftp browse with a single message was discussed but never fully implemented.)
Parent Group(s): BROWSE_REQUEST
Child Group(s): DATASET_ID, GRANULE_ID (sequence)
ODL Type: Aggregate

Keyword: BROWSE_ONLY
Synopsis: Indicates only granules having associated browse images should be returned from the inventory search.
Parent Group(s): INVENTORY_SEARCH
ODL Type: Symbol
Maximum Length: 1
Possible value(s): Y

Keyword: BROWSE_PRODUCT_DESCRIPTION
Synopsis: Browse product (image) description.
Parent Group(s): DATASET

ODL Type: Sequence String
Maximum Length: 80

Keyword: BROWSE_REQUEST
Synopsis: Message requesting transfer of browse image(s). In fact, no V0 system sends or requires USER_AFFILIATION in BROWSE_REQUESTS.
Child Group(s): BROWSE_TYPE, MESSAGE_ID, MONITOR group, CONTACT_ADDRESS group, BROWSE_GRANULES group, [AUTHENTICATOR], DATA_CENTER_ID, [ECS AUTHENTICATOR], [USER AFFILIATION group], VERSION group
ODL Type: Aggregate

Keyword: BROWSE_TYPE
Synopsis: Type of delivery for browse image. In a BROWSE_REQUEST, FTP_ONLY means "send ftp browse" and Y means "send integrated browse." In a GRANULE, N means "not available," FTP_ONLY means "available only as FTP," and Y means "available as both." There is currently no value for "available by integrated only."
Parent Group(s): BROWSE_REQUEST, GRANULE
ODL Type: Symbol

Maximum Length: 8
Possible value(s): Y | N | FTP_Only

Keyword: CAMPAIGN
Synopsis: Name(s) of campaign/project that gathered data associated with a dataset, granule, or search. For backward compatibility, a single campaign value uses a string, multiple use a sequence string.
Parent Group(s): DIRECTORY_SEARCH, DATASET, GRANULE,
INVENTORY_SEARCH
ODL Type: String or Sequence String
Maximum Length: 80

Keyword: CATEGORY
Synopsis: Affiliation category (USA or non-USA) for a user
Parent Group(s): USER_AFFILIATION
ODL Type: String
Maximum Length: 7
Possible value(s): USA, NOT USA

Keyword: CENTROID_LAT
Synopsis: Latitude of center point coordinate where coverage is described as a quadrilateral.
Parent Group(s): POLYGON_LOC group for INVENTORY_RESULTS
ODL Type: Real
Maximum Length: 9

Keyword: CENTROID_LON
Synopsis: Longitude of center point coordinate where coverage is described as a quadrilateral.
Parent Group(s): POLYGON_LOC group for INVENTORY_RESULTS
ODL Type: Real
Maximum Length: 9

Keyword: CITY
Synopsis: City of the address
Parent Group(s): BILLING_ADDRESS, CONTACT_ADDRESS, DAAC_CONTACT_ADDRESS, SHIPPING_ADDRESS,
ODL Type: String
Maximum Length: 30
Possible value(s): any string

Keyword: COMMENT
Synopsis: Arbitrary text information about corresponding granule, dataset, or package provided by the data center.
Parent Group(s): DATASET, GRANULE
ODL Type: Sequence String
Maximum Length: 60
Possible value(s): any string

Keyword: CONTACT_ADDRESS

Synopsis: The address portion of a user's contact information.
Parent Group(s): BROWSE_REQUEST, PRODUCT_REQUEST
Child Group(s): [TITLE], [ORGANIZATION], ADDRESS, USER_AFFILATION, CITY, EMAIL, [FAX], FIRST_NAME, [MIDDLE_INITIAL], LAST_NAME, PHONE, [STATE], [COUNTRY], [ZIP]
ODL Type: Aggregate

Keyword. CONTACT_NAME

Synopsis. Name of contact at the Data Center.

Parent Group(s). DAAC_CONTACT_ADDRESS

ODL Type. String

Maximum Length. 80

Possible value(s): any string

Keyword: COUNTRY
Synopsis: Country of the address.
Parent Group(s): SHIPPING_ADDRESS, BILLING_ADDRESS, CONTACT_ADDRESS, DAAC_CONTACT_ADDRESS
ODL Type: String
Maximum Length: 30

Keyword: DAAC_CONTACT_ADDRESS
Synopsis: User Support contact information (including DAAC order id) for one or more data sets. Information associated with one or more datasets. Group may be repeated in an FTP_BROWSE_RESULT or PRODUCT_RESULT, though older clients display only the first.
Parent Group(s): FTP_BROWSE_REQUEST group, PRODUCT_RESULT group
Child Group(s): [ADDRESS], CITY, CONTACT_NAME, COUNTRY, [EMAIL], [FAX], ORGANIZATION, PHONE, [STATE], [ZIP]

ODL Type: Aggregate

Keyword: DATA_CENTER_ID
Synopsis: Name of data center targeted by request or transmitting results.
Parent Group(s): DIRECTORY_RESULT, FTP_BROWSE_RESULT, INTEGRATED_BROWSE_RESULT, INVENTORY_RESULT, PRODUCT_RESULT, PRODUCT_REQUEST
ODL Type: Sequence String
Maximum Length: 10

Possible value(s): Must match (case blind) IMS valids name for DAAC.

Keyword: DATASET
Synopsis: Information about granules of a single data set in an inventory result. Data set information used to find directory information in GCMD.
Parent Group(s): DIRECTORY_RESULT, INVENTORY_RESULT

Child group(s): [COMMENT], [BROWSE_PRODUCT_DESCRIPTION], [PACKAGE_group], DATASET_ID, GRANULE group, [MD_ENTRY_ID], [NUMBER_OF_GRANULE_HITS],[DAY_NIGHT], [CAMPAIGN], [PARAMETER], [RESTRICTION], [SENSOR_NAME], [SOURCE_NAME], PROCESSING_LEVEL], STATUS_CODE, VALID_ACCOUNTS
ODL Type: Aggregate

Keyword: DATASET_ID
Synopsis: Names of valid IMS datasets associated with requests or results. Each value returned in INVENTORY_RESULTS and DIRECTORY_RESULTS must match (case blind) IMS valids name for some data set.
Parent Group(s): BROWSE_REQUEST, DATASET, [DIRECTORY_SEARCH], IMAGE, [INVENTORY_SEARCH]
ODL Type: Sequence String
Maximum Length: 80

Keyword: DAY_NIGHT
Synopsis: Flag requesting or specifying data gathered during daylight only or nighttime only. Support not provided for this field by all data centers for all data sets.
Parent Group(s): [GRANULE], [DATASET]
ODL Type: Symbol
Maximum Length: 1
Possible value(s): D | N

Keyword: DIRECTORY_RESULT
Synopsis: Provides result of directory level query against data center. V0 queries the GCMD for dataset information corresponding to the MD_ENTRY_IDs.

Child Group(s): DATA_CENTER_ID, DATASET group, MESSAGE_ID, MONITOR group, NUMBER_OF_DATASETS, STATUS_CODE, [STATUS_CODE_COMMENT], VERSION group
ODL Type: Aggregate

Keyword: DIRECTORY_SEARCH
Synopsis: Provides data for directory level search of data center

Child Group(s): [AUTHENTICATOR], [ECS_AUTHENTICATOR], [DATASET_ID], MESSAGE_ID, MONITOR group, [RANGE_LOC group], [CAMPAIGN], [PARAMETER], [SENSOR_NAME], [SOURCE_NAME], [START_DATE], [STOP_DATE], [PROCESSING_LEVEL], VERSION group
ODL Type: Aggregate

Keyword: EAST_LONGITUDE
Synopsis: Easternmost longitude for a range on the globe
Parent Group(s): RANGE_LOC

ODL Type: Real
Maximum Length: 9
Possible value(s): -180.0000 to +180.0000

Keyword: ECS_AUTHENTICATOR
Synopsis: Optional in every outgoing client message. Used for interfacing with ECS registration.
Parent Group(s): BROWSE_REQUEST, DIRECTORY_SEARCH, INVENTORY_SEARCH, PRODUCT_REQUEST, QUIT
ODL Type: String
Maximum Length: 100

Keyword: EMAIL
Synopsis: Internet e-mail address for associated person
Parent Group(s): BILLING_ADDRESS, CONTACT_ADDRESS, DAAC_CONTACT_ADDRESS, SHIPPING_ADDRESS
ODL Type: String
Maximum Length: 128
Possible value(s): any string

Keyword: ERROR
Synopsis: Data-Center provided text information about VALID_ACCOUNTS details.
Parent Group(s): VALID_ACCOUNTS

ODL Type: Sequence string
Maximum Length: 80

Keyword: EST_COST
Synopsis: Estimated cost of package, calculated from package's APPROX_COST.
Parent Group(s): LINE_ITEM
ODL Type: Real
Maximum Length: 16
Possible value(s): 0.0 to 9999999999999.99

Keyword: FAX
Synopsis: FAX phone number for associated person
Parent Group(s): BILLING_ADDRESS, CONTACT_ADDRESS, DAAC CONTACT ADDRESS, SHIPPING_ADDRESS
ODL Type: String
Maximum Length: 22
Possible value(s): any string

Keyword: FIRST_NAME
Synopsis: First name for addressed person.
Parent Group(s): BILLING_ADDRESS, CONTACT_ADDRESS, SHIPPING_ADDRESS
ODL Type: String
Maximum Length: 20
Possible value(s): any string

Keyword: FTP_BROWSE_RESULT
Synopsis: Results from a BROWSE_REQUEST for BROWSE_TYPE=FTP_ONLY.
Child Group(s): DAAC_CONTACT_ADDRESS group, DATA_CENTER_ID, MESSAGE_ID, MONITOR group, STATUS_CODE, [STATUS_CODE_COMMENT], TOTAL_FILE_SIZE, VERSION group
ODL Type: Aggregate

Keyword: GLOBAL_GRANULE
Synopsis: Granule has global coverage. Used in place of xxx_LOC keywords for granules with global coverage.
Parent Group(s): GRANULE
ODL Type: Symbol
Maximum Length: 1
Possible value(s): Y

Keyword: GLOBAL_GRANULES_ONLY
Synopsis: Only granules with global coverage should be returned in the result.
Parent Group(s): [INVENTORY_SEARCH]
ODL Type: Symbol
Maximum Length: 1
Possible value(s): Y

Keyword: GRANULE
Synopsis: Collection of metadata about a single data granule. For INVENTORY_RESULT, if SENSOR_NAME and SOURCE_NAME are not given in DATASET, they must be included in GRANULE.
Parent Group(s): DATASET
Child Group(s): [BROWSE_TYPE], GRANULE_ID, [PARAMETER], [PROCESSING_LEVEL], [SENSOR_NAME], [SOURCE_NAME], START_DATE, STOP_DATE, [CAMPAIGN], [COMMENT], [DAY_NIGHT], GLOBAL_GRANULE | POINT_LOC group | POLYGON_LOC group | RANGE_LOC group, [PACKAGE_ID]
ODL Type: Aggregate N/A

Keyword: GRANULE_ID
Synopsis: Granule's ID from Inventory
Parent Group(s): BROWSE_REQUEST, GRANULE, IMAGE
ODL Type: String
Maximum Length: 50
Possible value(s): any string

Keyword: GRANULE_LIMIT
Synopsis: Number of granules requested per data set
Parent Group(s): INVENTORY_SEARCH
ODL Type: Integer
Maximum Length: 10
Possible value(s): 1 to 2147483647

Keyword: IMAGE

Synopsis: Provides attributes of an integrated browse (image) file being transferred.
Parent Group(s): INTEGRATED_BROWSE_RESULT
Child Group(s): DATASET_ID, GRANULE_ID, IMAGE_ID, IMAGE_SIZE
ODL Type: Aggregate

Keyword: IMAGE_ID
Synopsis: Image identifier from Data Center
Parent Group(s): IMAGE group
ODL Type: String
Maximum Length: 50
Possible value(s): any string

Keyword: IMAGE_SIZE
Synopsis: Image size in bytes
Parent Group(s): IMAGE group
ODL Type: String
Maximum Length: 10
Possible value(s): 1 to 2147483647

Keyword: IMS_STAFF
Synopsis: Sent with every client message. Usually blank unless the client was run by a member of the IMS Staff. In the X client, copied from the IMS staff environment variable (as set in the shell script). In the Web gateway, set to "1" if the group "WWW_DEVELOPER" is present when loaded in the User Preferences.
Parent Group(s): VERSION
ODL Type: String
Maximum length: 10

Keyword: INFO_PROMPT
Synopsis: Data Center-supplied string to describe use of 'additional info' when ordering. Data center can provide usage or point to URL for help on ADDITIONAL_INFOs use.
Parent Group(s): PACKAGE
ODL Type: String
Maximum Length: 80

Keyword: INITIAL_USER_KEY
Synopsis: Original password used at the Data Center when first registering a user for data center-hosted clients. Set by shell for Data Center hosted clients. Originally intended to help user support groups distinguish individuals with similar names. May not be used much any more.
Parent Group(s): PRODUCT_REQUEST
ODL Type: String
Maximum Length: 12

Keyword: INTEGRATED_BROWSE_RESULT
Synopsis: Provides result of BROWSE_REQUEST where BROWSE_TYPE=Y.
Child Group(s): DATA_CENTER_ID, IMAGE group, MESSAGE_ID, MONITOR group, STATUS_CODE, [STATUS_CODE_COMMENT], [VERSION group]

ODL Type: Aggregate

Keyword: INVENTORY_RESULT
Synopsis: Provides result set from inventory query
Child Group(s): DATA_CENTER_ID, MESSAGE_ID, MONITOR group, [NUMBER_OF_DATASETS], STATUS_CODE, [STATUS_CODE_COMMENT], PACKAGE group, [DATASET group], [UNMAPPED_FIELD], [VERSION group]
ODL Type: Aggregate

Keyword: INVENTORY_SEARCH
Synopsis: Provides data to perform inventory query

Child Group(s):[AUTHENTICATOR], [ECS_AUTHENTICATOR], GRANULE_LIMIT, MESSAGE_ID, MONITOR group, [BROWSE_ONLY], [CAMPAIGN], [DATASET_ID], [DAY_NIGHT], GLOBAL_GRANULES_ONLY | POINT_LOC group | POLYGON_LOC group | RANGE_LOC group, [PARAMETER], [PROCESSING_LEVEL], [SENSOR_NAME], [SOURCE_NAME], [START_DATE], [START_DAY_OF_YEAR], [STOP_DATE], [STOP_DAY_OF_YEAR], VERSION group
ODL Type: Aggregate

Keyword: LAST_NAME
Synopsis: Last name for addressed person.
Parent Group(s): BILLING_ADDRESS, CONTACT_ADDRESS, SHIPPING_ADDRESS
ODL Type: String
Maximum Length: 20

Keyword: LATITUDE
Synopsis: Latitude for a point on the globe.
Parent Group(s): POINT_LOC, POLYGON_LOC
ODL Type: Sequence Real
Maximum Length: 8
Possible value(s): -90.0000 to +90.0000

Keyword: LINE_ITEM
Synopsis: Information needed for ordering a package.
Parent Group(s): PRODUCT_REQUEST
Child Group(s): [EST_COST], MEDIA_TYPE, [PACKAGE_ID], PROCESSING_OPTIONS, [MEDIA_FORMAT, [ADDITIONAL_INFO], [BILLING_ID], DATASET_ID]
ODL Type: Aggregate

Keyword: LONGITUDE
Synopsis: Longitude for a point on the globe.
Parent Group(s): POINT_LOC, POLYGON_LOC
ODL Type: Sequence Real
Maximum Length: 9
Possible value(s): -180.0000 to +180.0000

Keyword: MAP_PROJECTION_TYPE

Synopsis: Map projection type under which polygon was defined.
Parent Group(s): POLYGON_LOC
ODL Type: String
Maximum Length: 80
Possible value(s): PLATE_CARREE, NORTH_POLAR_STEREOGRAPHIC,
SOUTH_POLAR_STEREOGRAPHIC

Keyword: MD_ENTRY_ID
Synopsis: Global Change Master Directory entry (DIF) id
Parent Group(s): DATASET, GCMD_SEARCH
ODL Type: String
Maximum Length: 31
Possible value(s): any string

Keyword: MEDIA_FORMAT
Synopsis: For Parent = LINE_ITEM, Media format selected by user for order on this line item. For Parent = MEDIA_TYPE, description of distribution formats available for this package and media type.
Parent Group(s): LINE_ITEM, MEDIA_TYPE
Child Groupl(s): APPROX_COST, FORMAT_ID
ODL Type: For Parent = LINE_ITEM, String. For Parent = MEDIA_TYPE, Aggregate
Maximum Length: For Parent = LINE_ITEM, 30

Keyword: MEDIA_TYPE
Synopsis: For Parent=PROCESSING_OPTION, description of media on which package can be distributed. For Parent = LINE_ITEM, Medium selected by user for this line item.
Parent Group(s): PROCESSING_OPTION, LINE_ITEM
Child Group(s): TYPE-ID, NUMBER_OF_MEDIA_FORMAT, MEDIA_FORMAT group
ODL Type: For Parent = LINE_ITEM, String. For Parent=PROCESSING_OPTION, Aggregate
Maximum Length: For Parent = LINE_ITEM, 20

Keyword: MESSAGE_ID
Synopsis: Identifier used to track messages. Generated by the client or Web gateway.
Parent Group(s): BROWSE_REQUEST, DIRECTORY_RESULT, DIRECTORY_SEARCH, FTP_BROWSE_RESULT, INTEGRATED_BROWSE_RESULT, INVENTORY_RESULT, INVENTORY_SEARCH, PRODUCT_REQUEST, PRODUCT_RESULT
ODL Type: String
Maximum Length: 30
Possible value(s): any string

Keyword: MIDDLE_INITIAL
Synopsis: Middle initial of name for this address.
Parent Group(s): BILLING_ADDRESS, CONTACT_ADDRESS, SHIPPING_ADDRESS

ODL Type: String
Maximum Length: 1

Keyword: MONITOR
Synopsis: Statistics field for this message.
Parent Group(s): BROWSE_REQUEST, DIRECTORY_RESULT, DIRECTORY_SEARCH, FTP_BROWSE_RESULT, INTEGRATED_BROWSE_RESULT, INVENTORY_RESULT, INVENTORY_SEARCH, PRODUCT_REQUEST, PRODUCT_RESULT
Child Group(s): [RX_CLIENT], [RX_SERVER], TX_CLIENT, [TX_SERVER], [SESSION_ID]
ODL Type: Aggregate

Keyword: NORTH_LATITUDE
Synopsis: Northernmost latitude for a range on the globe.
Parent Group(s): RANGE_LOC
ODL Type: Real
Maximum Length: 8
Possible value(s): -90.0000 to +90.0000

Keyword: NUMBER_OF_DATASETS
Synopsis: Number of data sets being returned in query result set.
Parent Group(s): DIRECTORY_RESULT, INVENTORY_RESULT
ODL Type: Integer
Maximum Length: 10
Possible value(s): 1 to 2147483647

Keyword: NUMBER_OF_GRANULES
Synopsis: The number of granules included in the package.
Parent Group(s): PACKAGE
ODL Type: Integer
Maximum Length: 10
Possible value(s): 1 to 2147483647

Keyword: NUMBER_OF_GRANULE_HITS
Synopsis: Number of granules (or one more than the number) for this dataset being returned in full query result set. Should be present only in the last chunk of granules for a data set. Used to signal all granules have been returned. Some archives send value one greater than number actually returned to flag more granules were available than were returned.
Parent Group(s): DATASET
ODL Type: Integer
Maximum Length: 10
Possible value(s): 1 to 2147483647

Keyword: NUMBER_OF_OPTIONS
Synopsis: Number of PROCESSING_OPTION groups to follow.
Parent Group(s): PACKAGE

ODL Type: Integer
Maximum Length: 10
Possible value(s): 1 to 2147483647

Keyword: OPTION_ID
Synopsis: Description of processing option available for this package. In spite of the name, PROCESSING_OPTIONS is one OPTION_ID listed in one of the PROCESSING_OPTION groups.
Parent Group(s): PROCESSING_OPTION
ODL Type: String
Maximum Length: 30

Keyword: ORGANIZATION
Synopsis: Organization for address. On DAAC_CONTACT_ADDRESS this is a required field. On others, it is optional from profile.
Parent Group(s): CONTACT_ADDRESS, DAAC_CONTACT_ADDRESS, BILLING_ADDRESS
ODL Type: String
Maximum Length: 60

Keyword: PACKAGE
Synopsis: A collection of granules which can be ordered from an archive. All PACKAGE groups can be included before any DATASET group, or they may be intermixed; or PACKAGE groups may be included inside the corresponding DATASET groups. Package information in DATASET for DIRECTORY_RESULT is under investigation.
Parent Group(s): INVENTORY_RESULT
Child Group(s): DATA_CENTER_ID, DATASET_ID PACKAGE_ID, COMMENT, NUMBER_OF_GRANULES, NUMBER_OF_OPTIONS, PROCESSING_OPTION group, [INFO_PROMPT]
ODL Type: Aggregate

Keyword: PACKAGE_ID
Synopsis: Name of packages in which this granule can be ordered, or name of package being described or ordered. Special value of "*" allows a PACKAGE group to be used where granules can be ordered individually, all of which have the same characteristics. In such cases, the GRANULE_ID will be used as the PACKAGE_ID in the LINE_ITEM.
Parent Group(s): GRANULE, LINE_ITEM, PACKAGE
ODL Type: Sequence String
Maximum Length: 50
Possible values: Identifier or "*"

Keyword: PACKAGE_SIZE
Synopsis: String describing the size of the packages perhaps with units, e.g., "127 MB."

Parent Group(s): PROCESSING_OPTION
ODL Type: String
Maximum Length: 10

Keyword: PARAMETER
Synopsis: Geophysical term(s) associated with a dataset, granule, or search. Required for each GRANULE, but for INVENTORY_RESULT may be given instead at the DATASET level if it is the same for all GRANULEs in the DATASET.
Parent Group(s): DATASET, [DIRECTORY_SEARCH], GRANULE, [INVENTORY_SEARCH]
ODL Type: Sequence String
Maximum Length: 80

Keyword: PHONE
Synopsis: Voice telephone number of associated person.
Parent Group(s): BILLING_ADDRESS, CONTACT_ADDRESS, DAAC_CONTACT_ADDRESS, SHIPPING_ADDRESS
ODL Type: String
Maximum Length: 22
Possible value(s): any string

Keyword: POINT_LOC
Synopsis: Single point on the globe.
Parent Group(s): GRANULE, INVENTORY_SEARCH
Child Group(s): LATITUDE, LONGITUDE
ODL Type: Aggregate

Keyword: POLE_INCLUDED
Synopsis: Pole included in polygon (quadrilateral) region. Field omitted if neither pole included. Some clients cannot handle B (both poles).
Parent Group(s): POLYGON_LOC
ODL Type: Symbol
Maximum Length: 1
Possible value(s): N,| S

Keyword: POLYGON_LOC
Synopsis: Group of four latitude longitude pairs describing the search area or a granule's coverage. For PARENT = GRANULE, CENTROID_LAT and CENTROID_LON are required. For INVENTORY_SEARCH, TANGENT_LATITUDE, TANGENT_LONGITUDE and MAP_PROJECTION_TYPE are required.
Parent Group(s): GRANULE, INVENTORY_SEARCH
Child Group(s): [CENTROID_LAT], [CENTROID_LON], LATITUDE, LONGITUDE, [POLE_INCLUDED], [MAP_PROJECTION_TYPE], [TANGENT_LATITUDE], [TANGENT_LONGITUDE],

ODL Type: Aggregate

Keyword: PROCESSING_LEVEL
Synopsis: Level to which data has been processed associated with a dataset, granule, or search. Search can specify one or more; DATASET (for INVENTORY_SEARCH) or GRANULE normally specifies only single value. Usually ODL Type is Symbol for single value, Sequence String if more than one.
Parent Group(s): GRANULE, DATASET, DIRECTORY_SEARCH, INVENTORY_SEARCH
ODL Type: Symbol or Sequence String
Maximum Length: 2
Possible value(s): 0, 1, 1a, 1b, 2, 3, 4

Keyword: PROCESSING_OPTION
Synopsis: Repeating group listing options available for ordering this package. In spite of the name, PROCESSING_OPTIONS is the OPTION_ID listed in one of the PROCESSING_OPTION groups.
Parent Group(s): PACKAGE
Child Group(s): MEDIA_TYPE group, NUMBER_OF_MEDIA_TYPE, OPTION_ID, PACKAGE_SIZE
ODL Type: Aggregate

Keyword: PROCESSING_OPTIONS
Synopsis: Processing option selected by user for this LINE_ITEM. In spite of the name, PROCESSING_OPTIONS is the OPTION_ID listed in one of the PROCESSING_OPTION groups.
Parent Group(s): LINE_ITEM
ODL Type: String
Maximum Length: 30

Keyword: PRODUCT_REQUEST
Synopsis: Provides data for product request (order)
Child Group(s): [AUTHENTICATOR], [BILLING_ADDRESS group], CONTACT_ADDRESS group, DATA_CENTER_ID, [ECS_AUTHENTICATOR], [INITIAL_USER_KEY], LINE_ITEM group, MESSAGE_ID, MONITOR group, REQUEST_ID, [SHIPPING_ADDRESS group], USER_AFFILIATION group, VERSION group
ODL Type: Aggregate

Keyword: PRODUCT_RESULT
Synopsis: Order acknowledgement, including data center contact information. Only acknowledges receipt of order, not necessarily acceptance.
Child Group(s): DAAC_CONTACT_ADDRESS group, DATA_CENTER_ID, MESSAGE_ID, MONITOR group, STATUS_CODE, [STATUS_CODE_COMMENT], [VERSION group]

ODL Type: Aggregate

Keyword: PROTOCOL_VERSION
Synopsis: Version of message passing protocol, e.g., 3.5. Provided to allow changes in protocol that are not backward compatible. Since most changes have been made backward compatible, has not been used much.
Parent Group(s): VERSION
ODL Type: Real
Maximum Length: 10

Keyword: QUIT
Synopsis: Termination message. QUIT with STATUS_CODE 1000 used for ABORT.
Child Group(s): [AUTHENTICATOR], [DATA_CENTER_ID], [ECS_AUTHENTICATOR], MESSAGE_ID, MONITOR group, STATUS_CODE, [STATUS_CODE_COMMENT], VERSION group
ODL Type: Aggregate

Keyword: RANGE_LOC
Synopsis: Rectangular geographic range described by latitude and longitude boundaries.
Parent Group(s): [DIRECTORY_SEARCH], GRANULE, INVENTORY_SEARCH
Child Group(s): EAST_LONGITUDE, NORTH_LATITUDE, SOUTH_LATITUDE, WEST_LONGITUDE
ODL Type: Aggregate

Keyword: REQUEST_ID
Synopsis: Identifier assigned by the client for tracking a product request (order). Displayed on the X client's DAAC contact screen, followed by a slash and the DAAC_ORDER_ID if the PRODUCT_RESPONSE provides one.
Parent Group(s): PRODUCT_REQUEST
ODL Type: String
Maximum Length: 30

Keyword: RESTRICTION
Synopsis: Details of any ordering restrictions placed on the dataset.
Parent Group(s): DATASET group
ODL Type: Sequence String
Maximum Length: 60
Possible value(s): any string

Keyword: RX_CLIENT
Synopsis: Time stamp when the client received the server response. First string is integer number of seconds since Unix epoch; second optional string is integer number of microseconds since last integer second.
Parent Group(s): MONITOR group

ODL Type: Sequence String
Maximum Length: 20

Keyword: RX_SERVER
Synopsis: Time stamp when the server received the client request. First string is integer number of seconds since Unix epoch; second optional String is integer number of microseconds since last integer second.
Parent Group(s): MONITOR group
ODL Type: Sequence String
Maximum Length: 20

Keyword: SENDER_VERSION
Synopsis: String identifying the name and number of the sender (client or server) of the message. Used for documentation and debugging only.
Parent Group(s): VERSION
ODL Type: String
Maximum Length: 16

Keyword: SENSOR_NAME
Synopsis: Names of sensors associated with a dataset, granule, or search. SENSOR_NAME is required for each but for INVENTORY_RESULT, GRANULE, may be given instead at the DATASET level if it is the same for all GRANULEs in the DATASET.
Parent Group(s): GRANULE, DATASET, DIRECTORY_SEARCH, INVENTORY_SEARCH
ODL Type: Sequence String
Maximum Length: 30

Keyword: SHIPPING_ADDRESS
Synopsis: Address where requested data is to be sent.
Parent Group(s): PRODUCT_REQUEST
Child Group(s): [TITLE], [ADDRESS], [ORGANIZATION], CITY, [EMAIL], [FAX], FIRST_NAME, [MIDDLE_INITIAL], LAST_NAME, [PHONE], [STATE], COUNTRY, [ZIP]
ODL Type: Aggregate

Keyword: SESSION_ID
Synopsis: String identifying a user session. Used for matching purposes in SCRS statistics. Composed of the following, separated by colons: client host fully qualified domain name, client Unix process-id, session start date and time as YYMMDD:HHMMSS. In Web gateway, every communication with servers is a unique session (INVENTORY_SEARCH will have different SESSION_ID from PRODUCT_REQUEST).
Parent Group(s): MONITOR
ODL Type: String

Maximum Length: 86

Keyword: SOURCE_NAME
Synopsis: Name(s) of source/platform associated with a dataset, granule, or search. SOURCE_NAME is required for each GRANULE, but for INVENTORY_SEARCH may be given instead at the DATASET level if it is the same for all GRANULEs in the DATASET.
Parent Group(s): GRANULE, DIRECTORY_SEARCH, INVENTORY_SEARCH
ODL Type: Sequence String
Maximum Length: 30

Keyword: SOUTH_LATITUDE
Synopsis: Southernmost latitude for a range on the globe.
Parent Group(s): RANGE_LOC
ODL Type: Real
Maximum Length: 8
Possible value(s): -90.0000 to +90.0000

Keyword: START_DATE
Synopsis: Beginning date for search or granule temporal coverage.
Parent Group(s): GRANULE, DIRECTORY_SEARCH, INVENTORY_SEARCH
ODL Type: Date
Maximum Length: 20
Possible value(s): yyyy-mm-ddThh:mm:ss | yyyy-mm-ddThh:mm:ssZ

Keyword: START_DAY_OF_YEAR
Synopsis: Beginning day of seasonal interest. Query is for granules with start dates that are between START_DATE and STOP_DATE and are between START_DAY_OF_YEAR and END_DAY_OF_YEAR in whatever year. This could involve partial “seasons” (e.g., 2/1/93-2/28/95, days 1-90 would give Feb-Mar 93, Jan-Mar 94, Jan-Feb 95).
Parent Group(s): INVENTORY_SEARCH
ODL Type: Integer
Maximum Length: 3
Possible value(s): 1 TO 366

Keyword: STATE
Synopsis: US Postal state or foreign equivalent, if any, for address.
Parent Group(s): BILLING_ADDRESS, CONTACT_ADDRESS,
DAAC_CONTACT_ADDRESS, SHIPPING_ADDRESS
ODL Type: String
Maximum Length: 20
Possible value(s): any string

Keyword: STATUS_CODE
Synopsis: Numeric code giving status of query and/or server
Parent Group(s): DIRECTORY_RESULT, FTP_BROWSE_RESULT,
INTEGRATED_BROWSE_RESULT, INVENTORY_RESULT, PRODUCT_RESULT,
QUIT
ODL Type: Integer
Maximum Length: 4
Possible value(s): 1 to 20, or 1000

01 Successful query; query results returned
02 No match found
03 Data for selected source are not archived at DAAC
04 Data for selected sensor are not archived at DAAC
05 Data set is not archived at DAAC
06 Data for selected parameter(s) not archived at DAAC
07 Data for selected source, sensor, parameter(s) and/or data set are not archived at DAAC
08 Pertinent inventory system unavailable; try again later
09 Bad message; message contains syntax error(s)
10 Requested function not supported by this DAAC
11 System error, please try again later
12 Search too broad, narrow spatial and/or temporal search criteria
13 No data for selected campaign archived at DAAC; please reconstruct Search Query
14 “Browse_granules_only” selected, but no granules having browse data match
15 “Global_granules_only” selected, but no granules having global coverage match
16 No data for requested processing level at this DAAC, please reconstruct Search Query
17 Bad message; protocol error
18 System busy; try again later
19 System error; contact user support
20 Data not found due to spatial and/or temporal limitation
1000 User-requested abort of search

Keyword: STATUS_CODE_COMMENT
Synopsis: Free text comment provided by data center to further describe status. Some clients present this field in place of the fixed text associated with STATUS_CODE. Text should not be worded in a way that presumes the fixed text of the STATUS_CODE is presented as well.
Parent Group(s): DIRECTORY_RESULT, FTP_BROWSE_RESULT,
INTEGRATED_BROWSE_RESULT, INVENTORY_RESULT,
DIRECTORY_RESULT, FTP_BROWSE_RESULT,
INTEGRATED_BROWSE_RESULT
ODL Type: sequence string
Maximum Length: 256

Keyword: STOP_DATE
Synopsis: Ending date for search or granule temporal coverage.

Parent Group(s): GRANULE, DIRECTORY_SEARCH, INVENTORY_SEARCH
ODL Type: Date
Maximum Length: 20
Possible value(s): yyyy-mm-ddThh:mm:ss | yyyy-mm-ddThh:mm:ssZ

Keyword: STOP_DAY_OF_YEAR
Synopsis: Ending day of seasonal interest. Query is for granules with start dates that are between START_DATE and STOP_DATE and are between START_DAY_OF_YEAR and END_DAY_OF_YEAR in whatever year. This could involve partial “seasons” (e.g., 2/1/93-2/28/95, days 1-90 would give Feb-Mar 93, Jan-Mar 94, Jan-Feb 95).
Parent Group(s): INVENTORY_SEARCH
ODL Type: Integer
Maximum Length: 3
Possible value(s): 1 to 366

Keyword: TANGENT_LATITUDE
Synopsis: Current tangent (center) latitude of projection map.
Parent Group(s): POLYGON_LOC
ODL Type: Real
Maximum Length: 8
Possible value(s): -90.0000 to +90.0000

Keyword: TANGENT_LONGITUDE
Synopsis: Current tangent (center) longitude of projection map.
Parent Group(s): POLYGON_LOC
ODL Type: Real
Maximum Length: 9
Possible value(s): -180.0000 to +180.0000

Keyword: TITLE
Synopsis: Title for name of addressee.
Parent Group(s): CONTACT_ADDRESS, SHIPPING_ADDRESS, BILLING_ADDRESS
ODL Type: String
Maximum Length: 5

Keyword: TOTAL_FILE_SIZE
Synopsis: Combined uncompressed byte size of all FTP requests (may be exact or approximated).
Parent Group(s): FTP_BROWSE_RESULT
ODL Type: String
Maximum Length: 10

Possible value(s): 1 to 2147483647

Keyword: TX_CLIENT
Synopsis: Time stamp when client transmitted the request. First string is integer number of seconds since Unix epoch; second optional string is integer number of microseconds since last integer second.
Parent Group(s): MONITOR group
ODL Type: Sequence String
Maximum Length: 20

Keyword: TX_SERVER
Synopsis: Time stamp when server transmitted the response. First string is integer number of seconds since Unix epoch; second optional string is integer number of microseconds since last integer second.
Parent Group(s): MONITOR group
ODL Type: Sequence String
Maximum Length: 20

Keyword: TYPE
Synopsis: Affiliation categories: Government, Commercial, Academic, Other.
Parent Group(s): USER_AFFILIATION
ODL Type: String
Maximum Length: 15
Possible value(s): GOVERNMENT, COMMERCIAL, ACADEMIC, OTHER

Keyword: TYPE_ID
Synopsis: A valid value for media types for this PROCESSING_OPTION. A LINE_ITEM MEDIA_TYPE can be a TYPE_ID for the PROCESSING_OPTION MEDIA_TYPE group.
Parent Group(s): MEDIA_TYPE
ODL Type: String
Maximum Length: 30

Keyword: UNMAPPED_FIELD
Synopsis: Field(s) given in query but not used by the server in an inventory search.
Parent Group(s): INVENTORY_RESULT
ODL Type: Sequence String
Maximum Length: 80
Possible value(s): any keyword contained in the INVENTORY_SEARCH group

Keyword: USER_AFFILIATION
Synopsis: User’s self-classification from profile screen for statistics. Was once expected in BROWSE_REQUEST but never implemented there. Has been left as “optional” but no V0 system provides or expects it.
Parent Group(s): BROWSE_REQUEST, PRODUCT_REQUEST
Child Group(s): CATEGORY, TYPE
ODL Type: Aggregate

Keyword: VALID_ACCOUNTS
Synopsis: Accounting alternative for a dataset for this user. Omitted means accounting not required. If account required but the user has no valid account, then one VALID_ACCOUNTS group should be sent containing only the ERROR parameter with information to the user. Where multiple accounts are valid, the group is repeated, with each containing mandatory account number and optional balance and error fields.
Parent Group(s): DATASET
Child Group(s): [ACCOUNT_NUMBER], [BALANCE], [ERROR]
ODL Type: Aggregate

Keyword: VERSION

Synopsis: Message version information

Parent Groups: ACKNOWLEDGE, BROWSE_REQUEST, DIRECTORY_RESULT, DIRECTORY_SEARCH, FTP_BROWSE_RESULT, INTEGRATED_BROWSE_RESULT, INVENTORY_RESULT, INVENTORY_SEARCH, PRODUCT_CANCEL_REQUEST, PRODUCT_CANCEL_RESULT, PRODUCT_REQUEST, PRODUCT_RESULT, PRODUCT_STATUS_REQUEST, QUIT

Child Group(s): PROTOCOL_VERSION, SENDER_VERSION, [IMS_STAFF]

ODL Type: Aggregate

Keyword: WEST_LONGITUDE
Synopsis: Westernmost longitude for a range on the globe.
Parent Group(s): RANGE_LOC
ODL Type: Real
Maximum Length: 9
Possible value(s): -180.0000 to +180.0000

Keyword: ZIP
Synopsis: US Postal ZIP code or foreign equivalent for this address.
Parent Group(s): BILLING_ADDRESS, CONTACT_ADDRESS, DAAC_CONTACT_ADDRESS, SHIPPING_ADDRESS
ODL Type: String
Maximum Length: 15
Possible value(s): any string

A.2 ODL Keywords Used Only for Valids Transfer

The following ODL keywords are not used in the V0 message protocol but have been added to the keyword list because they are required for the V0 DAACs to supply dependent valids files as documented in Section 5.

Note: In a few cases (namely, DATASET, DATA_CENTER_ID, DATASET_ID) a keyword duplicates one used in the V0 message protocol, but a different description is applied when it is used for Valids transfer.

Keyword: BROWSE

Synopsis: Indicates what kinds of browse products are available from the archive for this data set. Used for documentation only. Not a part of dependent valids support files nor in V0 protocol messages.

Parent Group(s): DATASET

Child Group(s): FTP, INTEGRATED

ODL Type: Aggregate

Keyword: DATA_CENTER_ID

Synopsis: Data center whose dependent valids are being submitted.

Parent Group(s): VALIDS

ODL Type: String

Maximum length: 10

Keyword: DATASET

Synopsis: Contains dependent valids for a single data set being submitted by an archive.

Parent Group(s): VALIDS

Child Group(s): BROWSE, [CAMPAIGN], DATASET_COVERAGE, DATASET_ID, [DATE_AVAILABLE], [DAY_NIGHT_FLAG], [DEPENDENCY], FTP_PRODUCT_AVAILABLE, GRANULE_COVERAGE, MD_ENTRY_ID, [PARAMETER], PROCESSING_LEVEL, [SOURCE], [SENSOR]

ODL Type: Aggregate

Keyword: DATASET_COVERAGE

Synopsis: Spatial and temporal coverage of full data set. Used for documentation only. Not a part of dependent valids support files nor in V0 protocol messages.

Parent Group(s): DATASET

Child Group(s): SPATIAL, TEMPORAL

ODL Type: Aggregate

Keyword: DATASET_ID

Synopsis: Name of dataset whose dependent valids are being provided.

Parent Group(s): DATASET

ODL Type: String

Maximum length: 80

Keyword: DATE_AVAILABLE

Synopsis: Date (in no particular format) when dataset will become available. Not used in V0 protocol messages.

Parent Group(s): DATASET

ODL Type: String

Keyword: DAY_NIGHT_FLAG

Synopsis: Indicates whether dataset is wholly daytime data, nighttime, or both. Used for documentation only. Not a part of dependent valids support files nor in V0 protocol messages.

Parent Group(s): DATASET

ODL Type: Sequence String

Maximum length: 5

Possible value(s): “DAY”, “NIGHT”

Keyword: DEPENDENCY

Synopsis: Groups of parameters, sources, and sensors that are logically dependent within a dataset’s valids. Presence in a dependency group or valids DATASET group determines what values are compatible for client selection. Not used in V0 protocol messages.

Parent Group(s): DATASET

Child Group(s): [PARAMETER], [SENSOR], [SOURCE]

ODL Type: Aggregate

Keyword: EXTENDED_CRITERIA_AVAIL

Synopsis: Extended search criteria applicable to a dataset. Data centers provide EXTENDED_CRITERIA_AVAIL as a part of their valids submission to give CRITERIA_NAMEs of the EXTENDED_CRITERIA that may be used for each dataset. Not used in V0 protocol messages.

Parent Group(s): DATASET

ODL Type: Sequence String

Maximum Length: 80

Keyword: FTP

Synopsis: Indicates whether data center provides ftp browse for some granules of this dataset. Not used in V0 protocol messages.

Parent Group(s): BROWSE

ODL Type: String

Maximum Length: 5

Possible value(s): “yes”, “true”, “no”, “false”, “”

Keyword: FTP_PRODUCT_AVAILABLE

Synopsis: Indicates whether the archive delivers dataset products via ftp. Used for documentation only. Not a part of dependent valids support files nor in V0 protocol messages.

Parent Group(s): DATASET

ODL Type: String

Maximum length: 5

Possible values: “yes”, “true”, “no”, “false”, “”

Keyword: GRANULE_COVERAGE

Synopsis: Spatial and temporal coverage of individual granules of a dataset. Used for documentation only. Not a part of dependent valids support files nor in V0 protocol messages.

Parent Group(s): DATASET

Child Group(s): SPATIAL, TEMPORAL

ODL Type: Aggregate

Keyword: INTEGRATED

Synopsis: Indicates whether data center provides integrated browse for some granules of this dataset. Not used in V0 protocol messages.

Parent Group(s): BROWSE

ODL Type: String

Maximum Length: 5

Possible Values: “yes”, “true”, “no”, “false”, “”

Keyword: SENSOR

Synopsis: Names of sensors associated with a dataset. Not used in V0 protocol messages.

Parent Group(s): DATASET, DEPENDENCY

ODL Type: Sequence String

Maximum Length: 30

Keyword: SOURCE

Synopsis: Name(s) of source/platform associated with a dataset. Not used in V0 protocol messages

Parent Group(s): DATASET, DEPENDENCY

ODL Type: Sequence String

Maximum Length: 30

Keyword: SPATIAL

Synopsis: Free text used to describe spatial coverage of the dataset or granules. Used for documentation only. Not a part of dependent valids support files nor in V0 protocol messages.

Parent Group(s): DATASET_COVERAGE, GRANULE_COVERAGE

ODL Type: String

Maximum Length: 80

Keyword: TEMPORAL

Synopsis: Free text used to describe temporal coverage of the dataset or granules. Used for documentation only. Not a part of dependent valids support files nor in V0 protocol message.

Parent Group(s): DATASET_COVERAGE, GRANULE_COVERAGE

ODL Type: String

Maximum Length: 80

Keyword: VALIDS

Synopsis: Contains dependent valids for all data sets being submitted by a single archive for use in client support files. One or more valids groups (followed by a single line containing “END”) are used to build the dependency bitmap files that support the IMS clients. Alternatively, the VALIDS group itself can be omitted in file and just contents of the group (the DATA_CENTER_ID followed by one or more DATASET) can be included in the valids file submitted. Not used in V0 protocol messages.

Child Group(s): DATA_CENTER_ID, DATASET

ODL Type: Aggregate

Abbreviations and Acronyms

B0SOT	Release B.0 Search and Order Tool
CCB	Configuration Control Board
CCR	Configuration Change Request
CDRL	Contract Data Requirements List
DAAC	Distributed Active Archive Center
DBMS	Data Base Management System
DCN	Document Change Notice
ECS	EOSDIS Core System
EOS	Earth Observing System
EOSDIS	Earth Observing System Data and Information System
ESDIS	Earth Science Data and Information System
FQDN	Fully Qualified Domain Name
FTP	File Transfer Protocol
GCMD	Global Change Master Directory
GUI	Graphical User Interface
HDF	Hierarchical Data Format
HTML	HyperText Markup Language
HTTP	Hypertext Transport Protocol
I&T	Integration and Test
ICD	Interface Control Document
ID	Identifier
IDL	Interactive Data Language
IK	IMS Kernel
IMS	Information Management System
IP	Internet Protocol
IRD	Interface Requirements Document

JEST	JAVA Earth Science Tool
NCSA	National Center for Supercomputer Applications
ODL	Object Description Language
PID	Process Identifier
TBS	To Be Supplied
V0	Version 0
WAIS	Wide Area Information Server
WWW	World-Wide Web